Deep Multimodal Speaker Naming

Yongtao Hu
The University of Hong Kong
ythu@cs.hku.hk

Jimmy SJ. Ren
SenseTime Group Limited
rensijie@sensetime.com

Jingwen Dai
Xim Industry Inc.
dai@ximmerse.com

Chang Yuan
Lenovo Group Limited
yuanchang@lenovo.com

Li Xu
SenseTime Group Limited
xuli@sensetime.com

Wenping Wang
The University of Hong Kong
wenping@cs.hku.hk

ABSTRACT

Automatic speaker naming is the problem of localizing as well as identifying each speaking character in a TV/movie/live show video. This is a challenging problem mainly attributes to its multimodal nature, namely face cue alone is insufficient to achieve good performance. Previous multimodal approaches to this problem usually process the data of different modalities individually and merge them using handcrafted heuristics. Such approaches work well for simple scenes, but fail to achieve high performance for speakers with large appearance variations. In this paper, we propose a novel convolutional neural networks (CNN) based learning framework to automatically learn the fusion function of both face and audio cues. We show that without using face tracking, facial landmark localization or subtitle/transcript, our system with robust multimodal feature extraction is able to achieve state-of-the-art speaker naming performance evaluated on two diverse TV series. The dataset and implementation of our algorithm are publicly available online.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: Content Analysis and Indexing

Keywords

Speaker Naming; Multimodal; CNN; Deep Learning

1. INTRODUCTION

Identifying speakers, or speaker naming (SN), in movies, TV series and live shows is a fundamental problem in many high-level video analysis tasks, such as semantic indexing and retrieval [21] and video summarization [16], etc. As noted by previous authors [6], automatic SN is extremely challenging as characters exhibit significant variation of visual appearance due to changes in scale, pose, illumination, expression, dress, hair style, etc. Additional problems with video acquisition, such as poor image quality and motion

blur, make the matter even worse. Previous studies using only a single visual cue, such as face features, failed to generate satisfactory results.

Real-life TV/movie/live show videos are all multimedia data consisting of multiple sources of information. In particular, audio provides reliable supplementary information for SN task because it is closely associated with the video. In this paper, we propose a novel CNN based learning framework to tackle the SN problem. Unlike previous methods which investigated different modalities individually, our method automatically learns the fusion function of both face and audio cues and outperforms other state-of-the-art methods without using face/person tracking, facial landmark localization or subtitle/transcript. Our system is also trained end to end, providing an effective way to generate high quality intermediate unified features to distinguish outliers.

Contributions. 1) a novel CNN based framework which automatically learns high quality multimodal feature fusion functions; 2) a systematic approach to reject outliers for multimodal classification tasks typified by SN, and 3) a state-of-the-art system for practical SN applications.

2. RELATED WORK

Automatic SN in TV series, movies and live shows has received increasing attention in the past decade. In previous works like [11], SN was considered as an automatic face recognition problem. Recently, more researchers have tried to make use of video context to boost performance. Most of these works focused on *naming face tracks*. In [6], cast members are automatically labelled by detecting speakers and aligning subtitles/transcripts to obtain identities. This approach had been adapted and further refined by [15]. Bauml *et al.* [2] use a similar method to automatically obtain labels for those face tracks that can be detected as speaking. However, these labels are typically noisy and incomplete (i.e., usually only 20-30% of the tracks can be assigned a name) [2]. That is mainly due to that speaker detection relies heavily on lip movement detection, which is not reliable for videos of low quality or with large face pose variation.

In [17], each TV series episode is modeled as a Markov Random Field, which integrates face recognition, clothing appearance, speaker recognition and contextual constraints in a probabilistic manner. The identification task is then formulated as an energy minimization problem. In [19, 20], person naming is resolved by a statistical learning or multiple instances learning framework. Bojanowski *et al.* [4] utilize scripts as weak supervision to learn a joint model of actors and actions in movies for character naming. Although
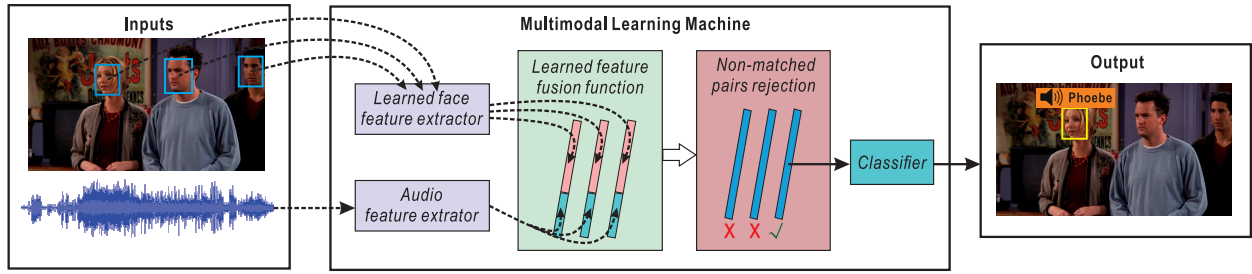
Figure 1: Multimodal learning framework for speaker naming.

these methods try to solve character naming or SN problem in new machine learning frameworks, they still heavily rely on accurate face/person tracking, motion detection, landmark detection and aligned transcripts or captions.

Unlike all these previous works, our approach does not rely on face/person tracking, motion detection, facial landmark localization or subtitle/aligned transcript as well as handcrafted features engineering. With only the input of cropped face regions and corresponding audio segment, our approach recognizes speaker in each frame in real-time.

3. MULTIMODAL CNN FRAMEWORK

Our approach is a learning based system in which we fuse the face and audio cues in the feature extraction level. The face feature extractor is learned from data rather than handcrafted. Then our learning framework is able to leverage both face and audio features and learns a unified multimodal feature extractor. This enables a larger learning machine to learn a unified multimodal classifier which takes both face image and speaker's sound track as inputs. The overview of the learning framework is illustrated in Figure 1.

3.1 Multimodal CNN Architecture

We adopted CNN [10] as the baseline model in our learning machine. As we will see shortly, CNN's architecture is inherently extensible. This makes our extension to multimodal learning concise, efficient but powerful.

The role of CNN in our framework is two-fold. Firstly, it learns a face feature extractor from face imagery data so that we have a solid face recognition baseline. Secondly, it combines both face feature extractor as well as the audio feature extractor and learns a unified multimodal classifier.

Figure 2 illustrates the design of our model. We will later show with insights that this model is very effective for SN tasks despite its conciseness.
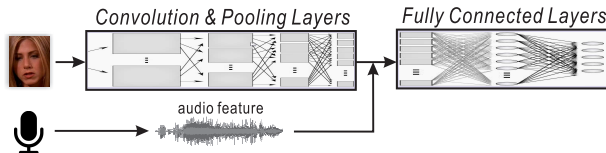

Figure 2: Multimodal CNN architecture.

In the trainable face feature extractor part, each layer of the network can be expressed as

$$N_c(\mathcal{I}) = \sigma(\mathcal{P}(\sigma(\mathcal{I} * K^l + b^l))), \; l = 1, 2, ..., n, \quad (1)$$

where $\mathcal{I}$ is the input for each layer. $\mathcal{I}$ is usually a 3D image volume, namely 3-channel input images when $l = 1$,

multi-channel feature maps when $1 < l \leq n$. K^l and b^l are the trainable convolution kernels and trainable bias term in layer l respectively. σ represents the nonlinearity in the network, which is modeled by a rectifier expressed as $f(x) = \max(0, x)$. $\mathcal{P}$ is a pooling function which subsamples the inputs by a factor of 2. Same nonlinearity is applied after the pooling function. When $l = n$ the output of $N_c(\mathcal{I})$ is a one dimensional high level feature vector.

For audio feature extraction, we use mel frequency cepstral coefficients (MFCCs) [14]. The MFCCs of one audio frame is also an one dimensional feature vector. This allows us to ensemble a unified multimodal feature by stacking $N_c(\mathcal{I})$ and MFCCs together.

It is worth noting that stacking of face feature and MFCCs in this stage is non-trivial in terms of classification. The reason is the ensuing trainable classifier essentially learns a higher dimensional nonlinear feature representation of the previous layer by mapping the stacked multimodal feature to a higher dimensional feature space. This is expressed as

$$N_f(\mathcal{F}) = \sigma(\mathcal{F} \cdot \mathcal{W}^l + b^l), \; l = n + 1, n + 2, ..., m, \quad (2)$$

where $\mathcal{F}$ is the stack of face feature and MFCCs with layer $l = n + 1$. When $n + 1 < l < m$, we impose constraint $Dim(\mathcal{F}^{l-1}) < Dim(\mathcal{F}^l)$ where $Dim()$ denotes the dimension of the intermediate feature vector, which promotes the learning of higher dimensional feature mapping. Such feature mapping is realized by the trainable weights $\mathcal{W}$ and b as well as the nonlinear σ. The system outputs the decision values of each class label by going through a softmax layer when $l = m$. The cross-entropy error function $\sum_{i=0}^{n} \ln(o_i) \cdot t_i$ is used as the error function during training, where o_i is the i-th element in $N_f(\mathcal{F})$, t is the ground truth class label. Though the conciseness of the model, one key insight of this approach is the whole system is trained end to end such that the influence of face feature extractor and MFCCs to the whole network is interwinding through learning.

Multimodal Feature Extraction. One important character of the CNN based classifier is its intermediate layers are essentially high level feature extractors. Previous studies [12] showed that such high level features is very expressive and can be applied in tasks such as recognition and content retrieval. It was not clear if such high level feature extraction mechanism works well in the context of multimodal learning. We will show in our experiments that our method is able to generate high quality multimodal features which is highly expressive in distinguishing outlier samples. This discovery forms one of the most important building blocks of making our system superior for real-life SN applications.

4. EXPERIMENTS

Experimental Setup. We evaluate our framework on over three hours videos of nine episodes from two TV series, i.e. "Friends" and "The Big Bang Theory" ("BBT"). For "Friends", faces and audio from *S01E03* (Season 01, Episode 03), *S04E04*, *S07E07* and *S10E15* serve as the training set and those from *S05E05* as the evaluation set. Note that, the whole "Friends" TV series of ten seasons is taken over a large time range of ten years. To leverage such a long time span, we intentionally selected these five episodes that spans the whole range. For "BBT", as in [17], *S01E04*, *S01E05* and *S01E06* are for training and *S01E03* for testing. For these two TV series, we only report performance of the leading roles, including six ones of "Friends", i.e. *Rachel, Monica, Phoebe, Joey, Chandler* and *Ross*, and five ones of "BBT", i.e. *Sheldon, Leonard, Howard, Raj* and *Penny*.

We conduct three experiments in terms of 1) face recognition; 2) identifying non-matched face-audio pairs and 3) real world SN respectively. For face recognition using both face and audio information, we only identify matched face-audio pairs. We further show how our model be able to classify matched face-audio pairs from non-matched ones. It is worth noting that the first two experiments provide solid foundations towards achieving promising performance in our third real world SN experiment. It also justifies the effectiveness of the building blocks in our resulting system.

Our CNN's detailed setting is described as follows. The network has 2 alternating convolutional and pooling layers in which the sizes of the convolution filters used are 15×15 and 5×4 respectively. The connection between the last pooling layer and the fully connected layer uses filters of size 7×5. The number of feature maps generated by the convolutional layers are 48 and 256 respectively. For fully connected layers, the number of hidden units are $1,024$ and $2,048$ respectively. Such architecture requires more than 11 million trainable parameters. All the bias terms are initialized to 0.01 to prevent the dead unit caused by rectifier units during training. All other parameters are firstly initialized within the range of -1 to 1 drawn from a Gaussian distribution and then scaled by the number of fan-ins of hidden unit they connect to. Average pooling of factor 2 is used throughout the network.

4.1 Face Model

We evaluate our model for face recognition on "Friends" (all face images resized to 50×40). We also test four previous algorithms under the same setting, i.e. Eigenface [18], Fisherface [3], LBP [1] and OpenBR/4SF [9].

Accuracies of these four previous methods are 60.7%, 64.5%, 65.6% and 66.1% respectively. All four previous algorithms fail to work well (all $< 70\%$), on the other hand, our method works better for every subject and achieves an accuracy of 86.7%. The results are expected as previous algorithms either require alignment of the face images or detecting facial feature points or both. This makes them not able to work well in the small sized face images that are extracted from unconstrained videos, which has no guarantee of alignment of the images, challenging large variations in pose, illumination and aging, etc.

We further apply audio to fine-tune our face model. The weights in this extended network is initialized by the parameters in the face-alone network. For the newly introduced parameters by the audio inputs, they are initialized in the
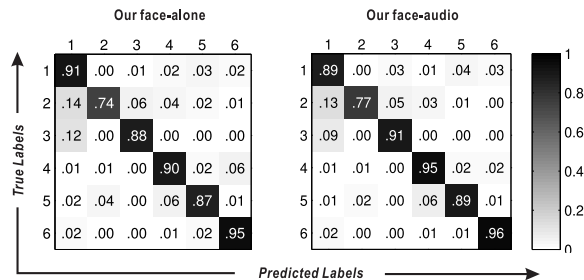

Figure 3: Confusion matrices of our face-alone and face-audio models for face recognition on "Friends". Labels 1-6 stand for the six subjects accordingly, i.e. *Rachel, Monica, Phoebe, Joey, Chandler* and *Ross*.

same way as presented before. Concerning audio features, a window size of 20ms and a frame shift of 10ms are used. We then select mean and standard deviation of 25D MFCCs, and standard deviation of 2-ΔMFCCs, resulting in a total of 75 features per audio sample. For each face, we catenate it with 5 audio samples of the same subject that are randomly selected to generate face-audio pairs.

Compared with previous face-alone model (acc: 86.7%), our face-audio model further improved this to 88.5% with corresponding confusion matrix shown in Figure 3. We can clearly see that, by adding audio information to the model, the accuracies of identifying all the subjects improve by 1-5% except a slight drop for *Rachel*.

4.2 Identifying Non-matched Pairs

In above experiments, all face-audio samples are matched pairs, i.e. belong to the same person. However, this condition cannot be fulfilled in practice. Consider a speaking frame, there are N faces, one of which is speaking, see Figure 1 as an example where $N = 3$. In order to find the correct speaker, we need to examine all face-audio pairs. All the pairs are non-matched except the one of the real speaker. And, it is almost impossible to train all possible non-matched pairs because new faces are unpredictable.

Thus, to identify non-matched pairs, a better way is to develop new strategies at the same time guarantee the quality of the face model. Instead of using the final output label of our face models, we explore the effectiveness of the features returned from the model in the last layer. As baseline, we train two binary supporting vector machines (SVM) [5]. One is trained on the 1024D fused feature that returned from our face-audio model and the other trained on 1024D face feature returned from our face-alone model concatenating with 75D audio feature (MFCC). We then train another SVM model using the same setting with the second SVM expect that we replace the 1024D face feature by the same dimensional fused feature from our face-audio model.

We test these three models on the evaluation video, which contains in total $17,131$ speaking frames. It will count as correct if the most confident face-audio pair matches, i.e. both from the same person. Two baseline SVMs achieve 82.2% and 82.9% respectively, whilst the third one can achieve 84.1%. Results clearly justify that the fused feature is more discriminative than the original face feature. On the other hand, we believe it also shows that the fused feature and MFCCs capture different but complimentary dimensions of the required information in distinguishing non-matched pairs.

Figure 4: Speaker naming result under various conditions, including pose (a)(c)(d), illumination (c)(d), small scale (b)(d), occlusion (a) and clustered scene (b)(d), etc (time stamp shown at the bottom left).

4.3 Speaker Naming

The goal of speaker naming is to identify the speaker in each frame, i.e. find out the matched face-audio pair and also identify it. It's worth noting that such a problem can be viewed as an extension of previous experiment of identifying non-matched pairs (reject all non-matched ones).

For the in total 17, 131 speaking frames in the evaluation video *Friends.S05E05*, we applied the third SVM to reject all non-matched pairs. The remaining pair will be assigned with the label returned by our face-audio model. Under such setting, we can achieve the SN accuracy of 90.5%. Sample SN result can be viewed from Figure 4.

Compared with Previous Works. Previous works [2, 17] have addressed similar SN problem by incorporating face, facial landmarks, cloth features, character tracking and associated video subtitles/transcripts. They evaluated on "BBT" and achieved SN accuracy of 77.8% and 80.8% respectively (evaluation on *S01E03*). In comparison, we can achieve SN accuracy of 82.9% without introducing any face/person tracking, facial landmark localization or subtitle/transcript.

4.4 Applications

Speaking activity is the key of multimedia data content. With our system, detailed speaking activity can be obtained, including speakers' locations, identities and speaking time ranges, etc, which further enables many useful applications. We highlight two major applications in the following (please refer to our supplementary video for details):

Video Accessibility Enhancement. With speakers' locations, we can generate on-screen dynamic subtitles next to the respective speakers thus enhance video accessibility for the hearing impaired [7] and enhance the overall viewing experience as well as reduce eyestrain for normal viewers [8].

Multimedia Data Retrieval and Summarization. With the detailed speaking activity, we can further achieve some high-level multimedia data summarization tasks, including characters conversation information and scene changing information, etc, based on which fast video retrieval is possible. We highlight such information in Figure 5.

5. CONCLUSIONS

In this paper, we propose a CNN based multimodal learning framework to tackle the task of *speaker naming*. Our approach is able to automatically learn the fusion function of both face and audio cues. We show that our multimodal learning framework not only obtains high face recognition
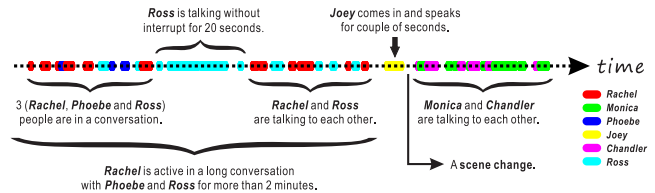

Figure 5: Speaking activity and video summarization for a 3.5 minutes video clip of *Friends.S05E05*.

accuracy but also extracts representative multimodal features which is the key to distinguish sample outliers. By combining the aforementioned capabilities, our system achieved state-of-the-art performance on two diverse TV series without introducing any face/person tracking, facial landmark localization or subtitle/transcript. The dataset and implementation of our algorithm, based on VCNN [13], are publicly available online at `http://herohuyongtao.github.io/research/publications/speaker-naming/`.

6. REFERENCES

[1] T. Ahonen, A. Hadid, and M. Pieti-kainen. Face description with local binary patterns: Application to face recognition. *TPAMI*, 2006.

[2] M. Bauml, M. Tapaswi, and R. Stiefelhagen. Semi-supervised learning with constraints for person identification in multimedia data. In *CVPR*, 2013.

[3] P. Belhumeur, J. Hespanha, and D. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *TPAMI*, 1997.

[4] P. Bojanowski, F. Bach, I. Laptev, J. Ponce, et al. Finding actors and actions in movies. In *ICCV*, 2013.

[5] C. Chang and C. Lin. LIBSVM: A library for support vector machines. *TIST*, 2011.

[6] M. Everingham, J. Sivic, and A. Zisserman. "Hello! my name is... Buffy" – automatic naming of characters in TV video. In *BMVC*, 2006.

[7] R. Hong, M. Wang, M. Xu, S. Yan, and T. Chua. Dynamic captioning: Video accessibility enhancement for hearing impairment. In *MM*, 2010.

[8] Y. Hu, J. Kautz, Y. Yu, and W. Wang. Speaker-following video subtitles. *TOMM*, 2014.

[9] J. Klontz, B. Klare, S. Klum, A. Jain, and M. Burge. Open source biometric recognition. In *BTAS*, 2013.

[10] A. Krizhevsky et al. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[11] C. Liu, S. Jiang, and Q. Huang. Naming faces in broadcast news video by image google. In *MM*, 2008.

[12] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *CVPR*, 2014.

[13] J. Ren and L. Xu. On vectorization of deep convolutional neural networks for vision tasks. In *AAAI*, 2015.

[14] M. Sahidullah and G. Saha. Design, analysis and experimental evaluation of block based transformation in mfcc computation for speaker recognition. *SC*, 2012.

[15] J. Sivic et al. "Who are you?" - learning person specific classifiers from video. In *CVPR*, 2009.

[16] K. Takenaka, T. Bando, S. Nagasaka, et al. Drive video summarization based on double articulation structure of driving behavior. In *MM*, 2012.

[17] M. Tapaswi, M. Bauml, and R. Stie-felhagen. "Knock! Knock! Who is it?" probabilistic person identification in TV-series. In *CVPR*, 2012.

[18] M. Turk and A. Pentland. Eigenfaces for recognition. *JCN*, 1991.

[19] J. Yang and A. Hauptmann. Naming every individual in news video monologues. In *MM*, 2004.

[20] J. Yang, R. Yan, and A. Hauptmann. Multiple instance learning for labeling faces in broadcasting news video. In *MM*, 2005.

[21] H. Zhang, Z. Zha, Y. Yang, S. Yan, et al. Attribute-augmented semantic hierarchy. In *MM*, 2013.