Gating Neural Network for Large Vocabulary Audiovisual Speech Recognition

Fei Tao, *Student Member, IEEE,* and Carlos Busso, *Senior Member, IEEE*

***Abstract*—*Audio-based automatic speech recognition* (A-ASR) systems are affected by noisy conditions in real-world applications. Adding visual cues to the ASR system is an appealing alternative to improve the robustness of the system, replicating the audiovisual perception process used during human interactions. A common problem observed when using *audiovisual automatic speech recognition* (AV-ASR) is the drop in performance when speech is clean. In this case, visual features may not provide complementary information, introducing variability that negatively affects the performance of the system. The experimental evaluation in this study clearly demonstrates this problem when we train an audiovisual state-of-the-art hybrid system with a *deep neural network* (DNN) and *hidden Markov models* (HMMs). This study proposes a framework that addresses this problem, improving, or at least, maintaining the performance when visual features are used. The proposed approach is a deep learning solution with a gating layer that diminishes the effect of noisy or uninformative visual features, keeping only useful information. The framework is implemented with a subset of the audiovisual CRSS-4ENGLISH-14 corpus which consists of 61 hours of speech from 105 subjects simultaneously collected with multiple cameras and microphones. The proposed framework is compared with conventional HMMs with observation models implemented with either a *Gaussian mixture model* (GMM) or DNNs. We also compare the system with a *multi-stream hidden Markov model* (MS-HMM) system. The experimental evaluation indicates that the proposed framework outperforms alternative methods under all configurations, showing the robustness of the gating-based framework for AV-ASR.**

***Index Terms*—Audiovisual large vocabulary automatic speech recognition, Multimodal deep learning, speech recognition.**

I. Introduction

A conventional problem in speech processing is *large vocabulary automatic speech recognition* (LVASR). Recently, advances in *deep neural networks* (DNNs) have resulted in improved acoustic models achieving outstanding performance [1], even approaching human level *word error rate* (WER) [2]. However, the background noise commonly observed in real world applications can impair the performance of *automatic speech recognition* (ASR) system. Therefore, there is still need for approaches to improve the robustness of *audio-based ASR* (A-ASR) system. Adding visual information describing lip movements is an appealing solution, creating *audiovisual ASR* (AV-ASR) systems. Studies have clearly demonstrated the audiovisual nature of speech perception [3]–[5], where lipreading improves speech intelligibility, especially in noisy environments. Motivated by these results, early work on AV-ASR demonstrated the benefit of using audiovisual solutions under noisy conditions [6], [7]. Since new portable devices usually have frontal cameras for teleconference, AV-ASR systems can be easily used in practical applications.

The fusion of audio and visual information is a critical problem in AV-ASR systems. Previous studies have explored several frameworks to capture the complementary information provided by each modality. The key challenge is to develop a system that improves, or at least maintains the performance of A-ASR across conditions. While it is relatively easy to demonstrate improved performance under noisy conditions [6]–[11], it is common to observe that some AV-ASR systems perform worse than A-ASR in acoustically clean environments [6], [9], [11], [12]. For example, previous evaluations showed that a DNN trained with concatenated audiovisual features achieved a lower performance than audio-only DNN systems [9], [13]. By concatenating the features, a DNN framework may fail in capturing the right coupling between the modalities. It is important to design a system that can automatically learn when to trust a feature, attenuating its effect when (1) the feature is not robust, or (2) other features are more discriminative.

This study presents a deep learning solution relying on a gating layer to deal with concatenated audiovisual features. Inspired by the gating mechanism utilized in the *long short term memory* (LSTM) framework [14], we introduce a new type of layer for deep learning referred to as a *gating layer* (GL). A GL consists of sigmoid units that filter out confusing information from the concatenated audiovisual features by multiplying its output with the output from a regular layer in the network. The results of these products are fed into the next layer, preventing irrelevant information to propagate. The neural network implemented with a GL is referred to as a *gating neural network* (GNN). The control process in a GNN is very similar to the gating mechanism in LSTM that controls the signal flow in *recurrent neural networks* (RNNs).

We conduct the evaluation of the proposed GNN framework with a subset of the audiovisual CRSS-4ENGLISH-14 corpus, which is one of the largest audiovisual corpora for audiovisual LVASR. The subset of the corpus used in this study has 61 hours of read and spontaneous speech from 105 participants. A key feature of this corpus is that the sessions were simultaneously recorded using two cameras (tablet, high resolution camera) and five microphones (two microphones in a smartphone, a microphone from a tablet, a desktop microphone and a close-talking microphone), allowing us to evaluate the GNN framework under different conditions. The experimental evaluation uses *hidden Markov models* (HMMs) to capture the dynamic nature of the speech. The observation models are implemented with *Gaussian mixture models* (GMMs), a DNN, and the proposed GNN. We also consider *multi-stream*

F. Tao, C. Busso are with the Department of Electrical and Computer Engineering, The University of Texas at Dallas, Richardson TX 75080, USA e-mail: fxt120230@utdallas.edu, busso@utdallas.edu

Manuscript received June 22, 2017; revised xx xx, 20xx.

hidden Markov models (MS-HMM). The results reveal the benefits of using the GNN-HMM framework under most of the conditions in clean and noisy recordings, achieving absolute improvements as high as 27.2% in WER over other methods.

The rest of the paper is organized as follows. Section II introduces the background and prior work related to this study. Section III introduces the CRSS-4ENGLISH-14 corpus used in this study. Section IV presents the proposed GNN framework for AV-ASR. Section V explains the experiments to evaluate the proposed approach, discussing the most important results. Finally, Section VI concludes the paper, describing potential extensions of the proposed solution.

II. Related Work

This section describes previous work on *visual-based ASR* (V-ASR) and AV-ASR systems, highlighting the limitations of current methods and the contributions of this paper.

A. Visual Speech Recognition System

V-ASR systems have recently emerged as an appealing solution for speech recognition, since studies have demonstrated that their performance does not significantly degrade under low *signal-to-noise ratio* (SNR) [6] or different speech modes [15]. The approaches differ in the features and the models used to recognize speech.

Unlike A-ASR systems, which commonly rely on *Mel-frequency cepstral coefficients* (MFCCs), there is no well-established feature vector for V-ASR systems. Several visual features have been proposed to find a good representation to characterize the orofacial movements associated with speech. These features can be broadly grouped into geometric and appearance-based features. Geometric features estimate distances between facial landmarks, especially from the lips [16], [17]. An advantage of geometric features is that they generalize better across speakers [18]. Appearance-based features estimate spatial or spectral information conveyed on a given *region of interest* (ROI). As the features depend on appearance, they are more sensitive to inter-speaker differences, video resolution, illumination conditions, and head pose variations. Examples include 2-D *discrete cosine transform* (DCT) coefficients, *active appearance models* (AAMs) [6]–[8], and local *histogram of oriented gradients* (HOG) [19]. Other studies rely on temporal features capturing the speech dynamic (e.g., optical flow [20]). In Tao and Busso [15], we proposed a feature set consisting of local DCT plus geometric distances of lip landmarks. The feature set combines geometric and appearance information, reducing speaker variability, which resulted in improved lipreading accuracy. The current trend across fields is to derive end-to-end systems where the features are automatically extracted from raw signals (i.e., in this case pixels around the orofacial area) [21]–[23]. An appealing example is the *convolutional neural network* (CNN) [24]. Noda et al. [25] claimed that using a raw image as the input of the system should improve the performance since it contains full information around mouth. However, the experiments only considered visemes rather than continuous speech. Our paper uses hand-crafted visual features, leaving as future work the use of CNN-based features. Since raw images will lead to high dimensional input, studies have relied on *deep bottleneck* (DB) features to compress the dimension of the feature vector. Petridis and Pantic [26] proposed to append DB features to the DCT coefficients extracted from the mouth area to achieve better performance. Tamura et al. [27] concatenated geometric and appearance features as inputs of a DNN system. They extracted DB features from this structure, which were the input vectors of a MS-HMM framework.

For modeling, *hidden Markov models* (HMMs) is normally used to capture the dynamic characteristic of the visual feature. Studies have also used *support vector machines* (SVMs) when the task is to recognize isolated phonemes or visemes [20]. For acoustic modeling, studies have used GMMs [6], *Neural networks* (NNs) [28], and DNNs [29]. Replacing GMMs with DNNs as the acoustic model in A-ASR has improved the performance [30], [31], since DNNs can find better feature representation for speech recognition [32]. A-ASR significantly outperforms V-ASR for high SNR as acoustic features are more discriminative for this task [6], [9]. In end-to-end systems, the temporal dynamic is commonly captured with *long short-term memory* (LSTM) units. End-to-end framework requires large dataset and powerful hardware platform. Petridis et al. [23] proposed a framework based on LSTM units to capture the temporal dynamic of a V-ASR system. However, the performance was not better than conventional systems based on HMMs. While our proposed system can be easily implemented with LSTM, the current version relies on HMMs.

B. Audiovisual Fusion for ASR

Katsaggelos et al. [33] presented a recent survey about audiovisual fusion, which emphasizes work on AV-ASR. This section focuses mostly on deep learning solutions. Advances in deep learning have facilitated principled approaches to fuse multiple modalities [9]. In most cases, deep learning solutions are used to obtain feature representations for each modality or joint feature representations that serve as inputs of an HMM to model the temporal evolution of the features. Ninomiya et al. [34] used a DB framework to extract acoustic and visual features. The DB audiovisual features were later concatenated and used as features of a GMM-HMM framework. Noda et al. [25] used CNN to extract visual feature and MFCCs as audio features. These bimodal input features were used as inputs of a MS-HMM system. On a Japanese audiovisual corpus consisting of 400 words, their framework to recognize isolated words was able to maintain recognition performance above other alternative frameworks as the SNR decreased. Huang and Kingsbury [11] used *deep belief networks* (DBNs) to build a continuous digit recognition system. They evaluated two configurations (1) using separate DBNs for audio and visual modalities, fusing the scores of their systems (2) creating a joint representation with independent layers for each modality which are then combined in a higher layer. The outputs are then used as either the input of a GMM-HMM framework or the observation model of an HMM, replacing the GMM. Ngiam et al. [9] stated that directly feeding concatenated audiovisual features into a DNN will decrease the performance compared with the DNN only trained with acoustic features. Instead of concatenating the features, they proposed several alternatives for fusing the two modalities with DNNs. One of the neural networks had two sub networks to find a good representation for each modality. The values from the sub networks were concatenated and fed into a neural network. The whole network was jointly tuned given the labels.

One of the main challenges in multimodal fusion described by Katsaggelos et al. [33] is creating dynamic weights for the modalities. For simplicity, the weights to combine the log-likelihood of the audiovisual streams are commonly fixed. As a result, the best configurations for noisy recordings are not ideal for acoustically clean environments. Creating dynamic weights to adjust the contributions of the modalities according to the given audiovisual features may overcome this problem. These systems can favor clean, robust and discriminative features. Gergen et al. [35] recently investigated this problem at the decoding level of the HMM. The study used a composite HMM, where the likelihood score was obtained by a weighted summation of the likelihood scores for the audio and video streams. During decoding, they introduced a cost function based on the ratio of the likelihood for the best path and the sum of the likelihoods for the N-best paths. The stream weight was determined by finding the optimum of the cost function for each input frame. Meutzner et al. [36] relied on *signal to noise ratio* (SNR) to estimate dynamic weights within a MS-HMM framework. However, SNR estimates may not be accurate enough to compute reliable weights. We believe a deep learning framework has the capability to directly learn the importance of the modalities, setting the corresponding parameters of the neurons. For example, Chung et al. [22] proposed two streams based on CNNs, one for audio and one for video, which were later combined using contrastive loss. Chung et al. [37] used two LSTMs, one for each modality, where the decoding was implemented with modality-dependent attention models. In our study, we learn the importance of each feature with the proposed gating layers allowing us to simply concatenate the audiovisual features.

C. Limitation of AV-ASR systems & Contributions of this Study

Most studies on AV-ASR have focused on digits or single-word recognition. These tasks are significantly simpler than LVASR [6], which reduces the potential application of AV-ASR. The studies have focused on improving ASR performance under noisy conditions [6] or different speech modes, such as whisper speech [15]. However, these AV-ASR systems with clean speech tend to provide lower performance than A-ASR [6], [9]. Large vocabulary audiovisual continuous speech recognition systems that perform better than, or at least equal to, A-ASR under all conditions are needed.

One of the key barriers in building AV-LVASR systems is the limited resources available in the community. Table I lists some of the largest audiovisual corpora. IBM collected a corpus consisting of 50 hours collected from 290 subjects (24,315 utterances with a vocabulary size of about 10,500 words) [6]. Huang and Kingsbury [11], also from IBM, used a corpus collected by a infrared headset consisting of 107 subjects speaking continuous digits (5.3 hours). However, these corpora are proprietary and have not been released to the research community. Some of the corpora contain small vocabulary tasks with limited lexical content such as the CUAVE [38], AV-GRID [39], and the OulusVS2 [40] databases. They do not provide suitable resources to train AV-LVASR using phoneme/viseme models. The TCD-TIMIT corpus [41] contains read speech, which is not as natural as spontaneous speech. There are only a few databases that are suitable for AV-LVASR (e.g., AusTalk [42] and AMI [43] corpora).

TABLE I: Summary of audiovisual corpora for ASR. The size is measured in terms of utterances (“utts”) or hours (“hrs”).

CORPUS	# SPEAKER	SIZE	TASK
IBM set from [6]	290	50 hrs	read ViaVoice scripts
IBM set from [11]	192	36 hrs	read digits and scripts
CUAVE [38]	36	7000 utts	connected digits
AV-GRID [39]	33	33,000 utts	small vocabulary
OuluVS2 [40]	52	1560 utts	read digits and phrases
TCD-TIMIT [41]	62	6913 utts	read TIMIT sentences
AusTalk [42]	$\sim$ 1000	3000 hrs	read&spontaneous speech
AMI [43]	?	100 hrs	group meeting
LRW [37]	?	1000 utts	spoken words in the wild
LRS [22]	?	118,116 utts	BBC videos

This study presents an AV-LVASR where visual features are carefully introduced in the system to maintain performance even when they do not provide discriminative information. In the worst case scenario, the system maintains the performance of an A-ASR system. The main contributions of this study are:

- Building an AV-LVASR system relying on a data collection effort that produced the CRSS-4ENGLISH-14 corpus – one of the largest audiovisual corpora introduced in the community.
- Proposing a new deep learning architecture to fuse audio-visual modalities based on gating layers that prevents noisy visual inputs from propagating in the network, improving WER under high and low SNR conditions.

III. Data Collection and Feature Extraction

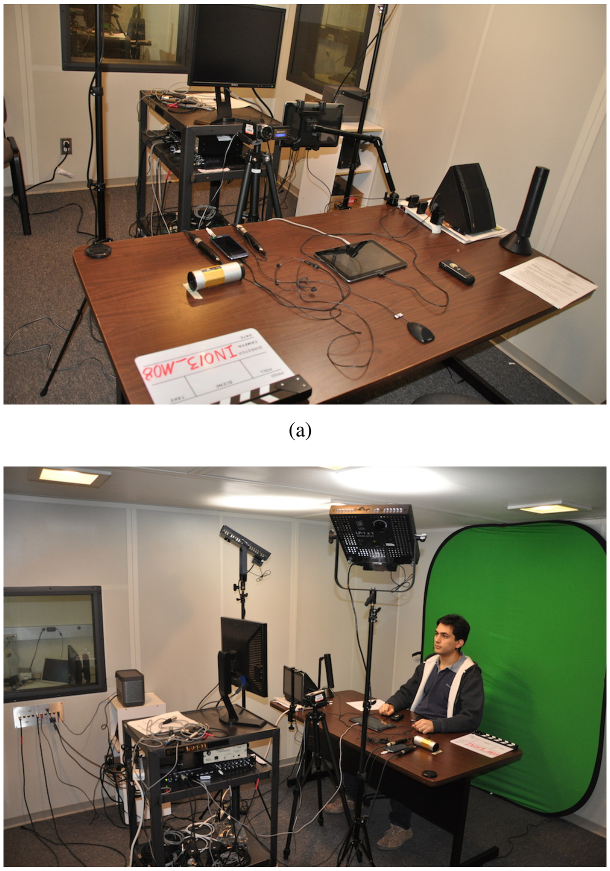
A. CRSS-4ENGLISH-14 corpus

This study uses the *CRSS-4English-14* corpus which was collected by the *Center of Robust Speech System* (CRSS) at *The University of Texas at Dallas* (UTDallas). The data was recorded in a $13ft \times 13ft$ *American Speech-Language-Hearing Association* (ASHA) certified sound booth. The booth was illuminated by two professional LED light panels (Figure 1). This corpus was recorded in English from 442 participants (217 female and 225 male speakers), with four English accents: American (115), Australian (103), Indian (112) and Hispanic (112). Table II lists the age statistics of the participants. We collected around 30 minutes per speaker, where we manually transcribed the data.

One of the key features of the corpus is the use of multiple microphones and cameras. Figure 1 shows the data collection settings. The audio was simultaneously collected with five microphones: a close-talking microphone (Shure Beta 53), a desktop microphone (Shure MX391/S), two channels of a cellphone (Samsung Galaxy SIII) placed on the desk, and a tablet (Samsung Galaxy Tab 10.1N) placed about two meters from

TABLE II: Age statistics of speakers in the CRSS-4English-14 corpus.

ACCENT	MAX	MIN	MEAN
Australian set	57	18	27.9
Hispanic set	48	18	23.3
Indian set	59	19	25.0
American set	59	18	26.1



(a)

(b)

Fig. 1: The equipment and setup used for the data collection.

the subject. All of the channels were set at a sampling rate of 44.1 kHz. The five channels are simultaneously collected with a digital recorder (Tascam US-1641). The video was simultaneously recorded with two cameras: a *high definition* (HD) camera (Sony HDR-XR100) and a camera from the tablet. The HD camera was set to 1440×1080 resolution at 29.97 frames per second, and the tablet was set to 1280×720 resolution at 24 frames per second. A green screen was placed behind the speaker to have a uniform background (Figure 1(b)). A monitor was placed in front of the participants to indicate the requested task (e.g., read prompted script, answer questions). The modalities were synchronized using a clapping board at the beginning of the recordings.

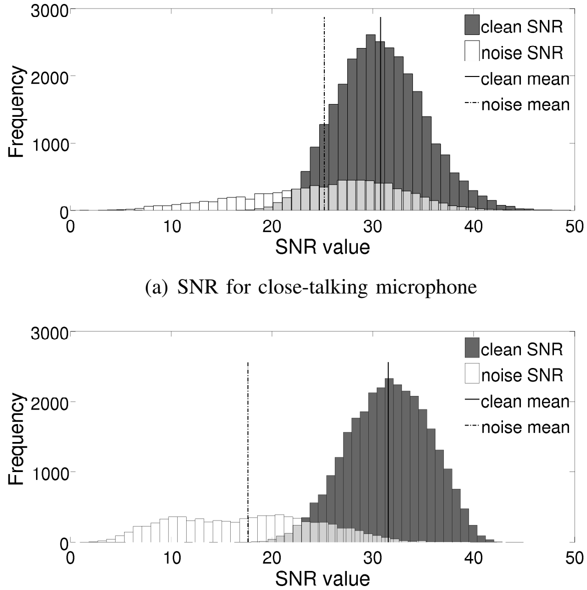
The first part of the data collection consisted of read and spontaneous speech. The read speech required the speaker to read prompted texts. The content included a variety of tasks such as continuous sentences (e.g., "I'd like to see an action movie tonight, any recommendation?"), questions (e.g., "How tall is the Mount Everest"), short phrases or commands (e.g., "change probe"), continuous numbers (e.g., "4, 3, 1, 8"), single words (e.g., "Worklist"), and cities (e.g., "Dallas, Texas"). The spontaneous speech required the speaker to respond to questions or hypothetical scenarios (e.g., "You are looking for suggestions on where to go on your next vacation. Give your vacation preferences and ask the system for suggestions"). We used an extensive set of prompted speech, questions, and scenarios for each of these tasks. Each participant was assigned a subset of these tasks, providing a rich lexical content. This part of the data collection includes only clean speech.

The second part of the data collection consists of noisy conditions. Previous studies have artificially added noise to the clean recordings to simulate noisy conditions. We use an audio speaker (Beolit 12) in the sound booth to play four types of prerecorded noises: mall, home, office and restaurant. This part of the recording is only five minutes, restricting the collection to read speech. We randomly selected slides for the read speech portion under clean conditions used in the given session, asking the participants to repeat the task under noisy conditions. The participants were free to move their head while completing the task.

This study only uses data from the American participants to reduce the accent effect. We observed problems with the videos of 10 subjects, so we use data from 105 participants (55 female and 50 male speakers). The total duration of the set is 60 hours and 48 minutes. We use videos from HD and tablet cameras. We only use two microphones to simplify the experimental evaluation: the close-talking microphone and the microphone from the tablet. Notice that the location of the tablet approximates the scenario of an individual using the tablet for teleconference, moving the technology closer to real-word applications. The data was randomly assigned to the train (70 speakers), validation (10 speakers) and test (25 speakers) sets. All the partitions are gender balanced. For comparison, the test data only contains the speech spoken in both clean and noisy sessions. The duration of the test data in clean conditions is around 3.1 hours; the duration of test data in noisy conditions is around 2.9 hours.

B. SNR Analysis for Close-Talking and Tablet Microphone

To understand better the noise level in the signal, we estimate the SNR for the clean and noisy conditions using the NIST speech SNR tool [44]. Figure 2 shows the histograms of SNR for the close-talking and tablet microphones for all the sentences used in this study. The close-talking microphone was placed about 1 cm from the mouth, so it was not significantly affected by the noise. This observation is clear in Figure 2(a), which shows overlap in the SNR histograms for clean and noisy conditions. However, the tablet was placed closer to the audio speaker producing the noise, so the SNR for its noisy condition is lower than the SNR for the close-talking channel. Therefore, we observe differences in the SNR histograms between clean and noisy conditions for the tablet channel (Figure 2(b)). This analysis is important to interpret the results presented in Section V.



(a) SNR for close-talking microphone

(b) SNR for tablet microphone

Fig. 2: The SNR distribution for clean and noisy conditions for the data collected with the two microphones.

C. Feature Extraction

We extract 13 MFCCs as our acoustic features. We downsample the speech to 16 kHz, estimating the features using a window size of 25 ms with 15 ms of overlap (i.e., 100 frames per second).

The visual features correspond to the set proposed in Tao and Busso [15]. This set contains local DCT around the lip plus geometric features. Figure 3 shows the flow chart used to extract visual features. First, we manually select a frame in which a subject is facing the camera with a neutral frontal pose, setting this frame as a template (Figure 3). We automatically estimate 49 facial landmarks from the frames and the template using the IntraFace toolkit [45]. The next step consists of normalizing the face. For this purpose, we estimate an affine transformation between each frame and the template. The affine transformation matrix is then applied to the frames to normalize the input frame, compensating for head rotations and face sizes. The estimation is based on the rigid points mainly located around the nose that do not move as a result of speech articulation (green circles in Figure 3). By using the same template across speakers, we create consistent poses, reducing the variability across frames. After the normalization, we define the *region of interest* (ROI). We estimate five geometric distances between six lip landmarks (Figure 3). The lip landmarks are also used to define a region inside the mouth from where we estimate 25 DCT coefficients. This 25D feature vector is referred to as *local DCT*. These features are motivated by the evaluations conducted on our previous work on lipreading [15] and audiovisual *voice activity detection* (VAD) [46]–[48], where using local DCT provided superior performance over DCT coefficients extracted over the entire mouth. We hypothesize that local DCTs are less sensitive to appearance differences across speakers (e.g., beard). The geometric and appearance based features are concatenated to create a 30D vector per frame that is used as the visual features. To synchronize the acoustic and visual features, we interpolate the visual features to get 100 frames per second.

IV. Proposed Approach

This section introduces the proposed *gating neural network* (GNN), which is used as the observation model of an HMM, creating a hybrid GNN-HMM system. This framework will be compared with observation models implemented with a DNN and GMMs. For consistency, the temporal modeling in all the conditions is implemented with standard HMMs. We use 43 phonemes, forming 2,641 senones.

A. Motivation

Before introducing the proposed approach, this section describes the baseline systems using either GMMs or DNNs as observation models. To demonstrate the problems with these models when we concatenate audio features with unreliable visual features, we present evaluations where visual features are replaced with random values (i.e., extreme case). This evaluation serves as a motivation of the proposed framework.

The first baseline is a *multi-stream HMM* (MS-HMM), which is a common framework for AV-ASR in previous studies [6], [25], [27], [33], [34]. It uses separate GMM-HMMs for each modality, combining the respective log-likelihood using Equation 1 [49] ($\mathbf{o}(t)$ is the audiovisual observation, $\mathbf{o_a}(t)$ is the acoustic observation, $\mathbf{o_v}(t)$ is the visual observation, s is the audiovisual state, s_a is the acoustic state and s_v is the visual state). For simplicity, the weights α_a and α_v are set to one. The GMM-HMM models were implemented with the HTK toolkit [50] using 32 mixtures per state.

$$\ln \mathcal{L}(\mathbf{o}(t)|s) = \alpha_a \ln \mathcal{L}(\mathbf{o_a}(t)|s_a) + \alpha_v \ln \mathcal{L}(\mathbf{o_v}(t)|s_v) \quad (1)$$

The second baseline model is an AV-ASR system implemented with a GMM-HMM framework, which we train using standard procedure [51]. The input to this system is just the concatenation of audio and visual features plus the first and second derivatives (i.e., 129D feature vector).

The third baseline system is the hybrid DNN-HMM framework [30], which has provided competitive performance in A-ASR. Figure 4(a) shows the architecture of the system, with four hidden layers. Using the pre-trained GMM-HMM, we run forced alignment on the training data to get senone ID for each feature frame. As commonly implemented in A-ASR [1], [52], [53], we use 15 contextual frames by combining the previous seven frames and the future seven frames for both audio and visual features (i.e., $[13+30]\times 15$). The label is the senone ID of the middle frame. The DNN is trained with this contextual feature vector using back propagation implemented with *Adagrad* [54] within each mini-batch [24]. During the testing stage, the input features are fed into the trained DNN to compute the posterior probability of each senone. We estimate the likelihood of the input feature by estimating the prior probability of the senones from their counts in the training data. The likelihood is then sent to the HMM for recognition.

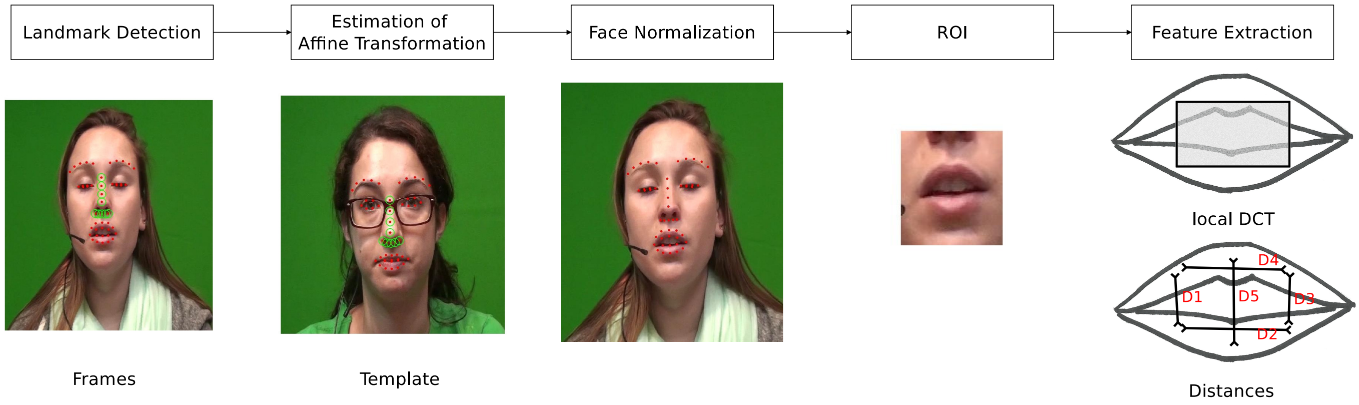
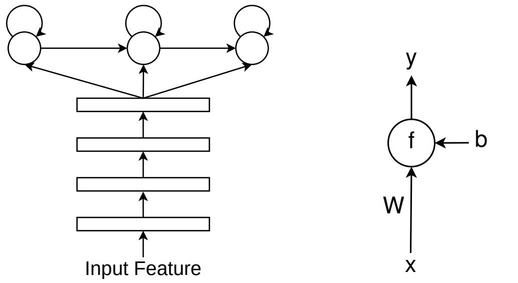


Fig. 3: The flowchart to extract visual features. We select a frame as a global template, which is used to consistently normalize all the frames. Green circles are used as rigid points for normalization. The rigid points are the less sensitive markers to changes in facial expression, which are used to normalize the size and orientation of the face. Facial landmarks are used to estimate geometric and appearance based features, creating a 30D feature vector.



(a) DNN-HMM System (b) Unit in DNN

Fig. 4: DNN-HMM system and the basic unit in DNN

TABLE III: Motivation of the proposed approach. Performance of a system trained with (1) audio features, and (2) audio features concatenated with random values ($*$ indicates one method is significantly better than others for a given condition).

MODEL	WER(%)
Audio GMM-HMM	4.62
Audio DNN-HMM	**3.70**
Audio GNN-HMM	4.00
Audio+Random MS-HMM	30.34
Audio+Random GMM-HMM	33.09
Audio+Random DNN-HMM	11.03
Audio+Random GNN-HMM	**4.52**$*$

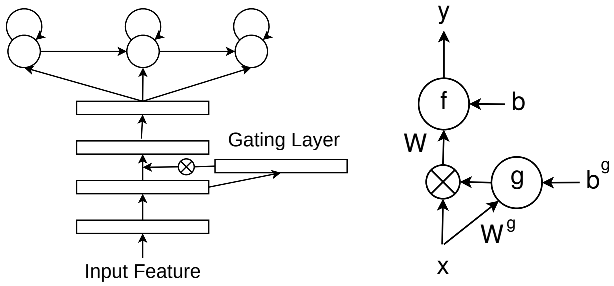
The basic unit in a DNN is shown in Figure 4(b). Using matrix form, the feed-forward computation follows Equation 3, where $\mathbf{x}$ is the output vector from the lower layer, $f()$ is the activity function, $\mathbf{W}$ is the weight associated with the connection between the lower and upper layers, and $\mathbf{b}$ is the bias. During back-propagation, the weight is updated following Equation 2 [24], [55], where η is the learning rate, $\mathbf{W}_i$ is the weight in the i_{th} iteration, and $\mathbf{e}$ is the back-propagating error from the upper layer. To iteratively update the weights, the partial derivative in Equation 2 is estimated in Equation 4. Several partially differentiable functions can be used including sigmoid, tanh, and maxout [56]. Our models are implemented with maxout with three nodes per unit.

$$\mathbf{W}_i = \mathbf{W}_{i-1} - \eta \frac{\partial \mathbf{e}}{\partial \mathbf{W}_{i-1}} \tag{2}$$

$$\mathbf{y} = f(\mathbf{Wx} + \mathbf{b}) \tag{3}$$

$$\frac{\partial \mathbf{y}}{\partial \mathbf{W}} = f'(\mathbf{Wx} + \mathbf{b})\mathbf{x} \tag{4}$$

We evaluate the performance of the A-ASR using the GMM-HMM and DNN-HMM frameworks on the validation set. For GMM, we have 25,000 mixtures in total, following the standard setup provided by the toolkit Kaldi [52]. For the DNN, we have four layers with 1,024 nodes in each layer. Table III gives the results. The performance for the GMM-HMM is 4.62%, and for the DNN-HMM is 3.70%. We are interested in evaluating the performance of an AV-ASR when the visual features provide little or no information. For analysis purposes, we replace visual features with random numbers drawn from a normal distribution, removing the correlation between the features and the phonetic unit. The dimension of the random feature vector is the same as the visual feature (i.e., 30 × 15 = 450D). We concatenate the audio features with random numbers and we re-run the evaluation. The performance drops to 33.09% for the GMM-HMM framework, and 11.03% for the DNN-HMM framework. We conducted the McNemar's test to evaluate if the differences in WER are statistically significant, asserting significance at p-value=0.01. The drop in performance is significant for both models (p-value $<$0.01). The WER for MS-HMM is 30.34%, which is lower than the GMM-HMM system with only acoustic features. While this analysis corresponds to an extreme case where the additional features are not discriminative, it illustrates the problem of using these frameworks for AV-ASR. This result motivates us to explore a new approach for filtering noisy features using a data-driven framework.



(a) GNN-HMM System (b) Unit in GNN

Fig. 5: Proposed GNN-HMM system which relies on gating layers.

B. Gating Neural Network (GNN) for AV-ASR

This section introduces the proposed GNN, shown in Figure 5(a), which filters non discriminative information, addressing the problem of concatenating audiovisual features for speech recognition. This goal is achieved by using a gating layer unit, described in Figure 5(b), which shows one connection between the gate unit and the input. The gate unit (the circle with f) is connected to all the input nodes. The output from the lower layer is point-wise multiplied with the output from the gating layer, sending the results as input to the upper layer (i.e., Hadamard product). The activity function $f()$ of the gating layer in Figure 5(b) is a sigmoid function, in which its output ranges between 0 and 1. If the output of the gating layer is 0, the multiplication result will be zero and the output from the lower layer will not propagate to the upper layer. If the output of the gating layer is 1, however, the gating unit responds equivalent to the unit in Figure 4(b). Therefore, the point-wise multiplication works as a "gate" to control whether the information goes through the network. The gating layer learns from the training data to filter the less discriminative information. In the extreme case where visual features are unreliable, the gating layer should attenuate all the visual features, relying only on the speech features.

The weight $\mathbf{W}$ in Figure 5(b) is learned with back propagation using Equation 2. The variable $\mathbf{W}^g$ is the weight connecting the input and the gating unit, which is learned by substituting the weight $\mathbf{W}$ with $\mathbf{W}^g$ in Equation 2, as shown in Equation 5. The key change in the training is in Equation 3, where we need to incorporate the point-wise multiplications in the feed forward computation ($\odot$ represents Hadamard product). Equation 6 gives the updated output vector $\mathbf{y}$, where $g()$ is the activity function of the gating unit. This study implements $f()$ with a sigmoid function. Two partial derivatives of $\mathbf{y}$ are required to learn the weights, one with respect to $\mathbf{W}$ (Equation 7) and the other with respect to $\mathbf{W}^g$ (Equation 8).

$$\mathbf{W}_i^g = \mathbf{W}_{i-1}^g - \eta \frac{\partial \mathbf{e}}{\partial \mathbf{W}_{i-1}^g} \tag{5}$$

$$\mathbf{y} = f(\mathbf{W}(g(\mathbf{W}^g\mathbf{x} + \mathbf{b}^g) \odot \mathbf{x}) + \mathbf{b}) \tag{6}$$

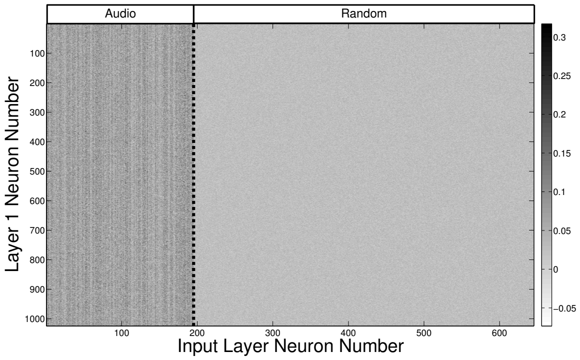


Fig. 6: The product of the gating neuron activity and connection weight between the input and the first hidden layer. The horizontal axis represents the number assigned to the input features and the vertical axis represents the node number in the first hidden layer.

$$\frac{\partial \mathbf{y}}{\partial \mathbf{W}} = f'(\mathbf{W}(g(\mathbf{W}^g\mathbf{x} + \mathbf{b}^g) \odot \mathbf{x}) + \mathbf{b}) \times (g(\mathbf{W}^g\mathbf{x} + \mathbf{b}) \odot \mathbf{x}) \tag{7}$$

$$\frac{\partial \mathbf{y}}{\partial \mathbf{W}^g} = f'(\mathbf{W}(g(\mathbf{W}^g\mathbf{x} + \mathbf{b}^g) \odot \mathbf{x}) + \mathbf{b}) \times \mathbf{W}(g'(\mathbf{W}^g\mathbf{x} + \mathbf{b}^g) \odot \mathbf{x})\mathbf{x} \tag{8}$$

We also evaluate the performance of the GNN-HMM framework when we replace the visual features with random numbers. We also use four layers, each of them with 1,024 nodes. The gating layer is added at the input layer taking the concatenated vector formed with audio features and random features (Sec. V-A). Table III shows the results of the evaluation. When we use only audio, the WER for the proposed GNN-HMM is slightly worse than the DNN-HMM framework decreasing the performance from 3.70% to 4.00% (p-value<0.01). However, the table clearly shows the benefits of the proposed framework when we concatenate audio and random features. The performance of the network only drops to 4.52%, in spite of the random inputs. This represents a relative improvement of 59% (p-value<0.01) over the DNN-HMM framework. This evaluation shows that the gating layers can attenuate the effect of non-discriminative features, reducing the activations propagated to the next layers.

Figure 6 visualizes the influence of the gating layers in the network. It shows the *mean absolute product* (MAP) of the activity of the gating layer and the weight associated with the connection between the input and the first hidden layer. The horizontal axis represents the input layers, where the first 195 dimensions are associated with audio features and the remaining 450 dimensions are associated with the random features. The vertical axis represents the neurons in the first hidden layer. The MAP is computed with Equation 9, where $\mathbf{x}_n$ are input samples in one mini-batch, which are randomly selected, and N is the mini-batch size (N=256). W is a $J \times I$

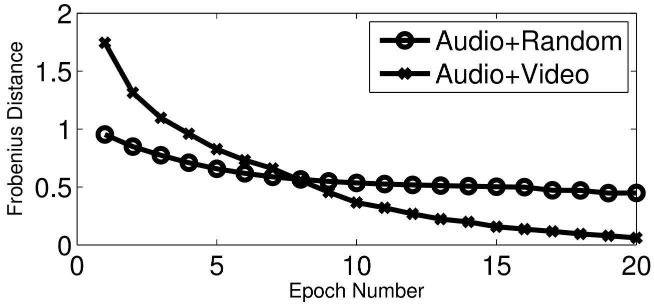


Fig. 7: The Frobenius distance of matrix $\mathbf{W}^g$ between consecutive epochs. The decreasing trend of this distance implies that the parameters are converging.

weight matrix, representing the weights between the I inputs and J neurons in layer 1. $g(\mathbf{W}^g\mathbf{x}_n + \mathbf{b}^g)$ is a row vector of dimension $1 \times I$, and $repmat$ creates a $J \times I$ matrix by repeating this row vector J times. This metric describes how a given input affects the neural network. When the product is small, the corresponding input will be scaled down and have a small impact on the neural network. The figure shows that the relative values for the acoustic features are consistently greater than the ones for the random features (i.e., darker colors). Figure 6 shows that the gating layers filter random features, maintaining the WER under 5%, even in this extreme case. Interestingly, all the parameters are learned from the data providing a principled framework for AV-ASR systems that are built with concatenated features.

$$MAP = \frac{1}{N}\sum_{n=1}^{N} |repmat(g(\mathbf{W}^g\mathbf{x}_n + \mathbf{b}^g)) \circ \mathbf{W}| \qquad (9)$$

C. Convergence of Gating Parameters

We evaluate the convergence behavior of the parameter $\mathbf{W}^g$ in the gating layer. We consider two cases where the acoustic features are concatenated with either random or visual features. $\mathbf{W}^g$ is a matrix with the weight coefficients connecting input and gating layers. We quantify the variations across epochs by estimating the Frobenius distance of $\mathbf{W}^g$ across consecutive epochs. Figure 7 provides the results over the first 20 epochs. The distance decreases as the number of epochs increases. This result implies that the parameters are converging, since the updates are smaller than in previous epochs. It can also be observed that the network with audiovisual inputs converges faster than the one with audio and random inputs. We hypothesize that real visual features facilitate the training of the gating layers.

D. Comparison Between GNN and Other Emerging Deep Learning Techniques

This section compares the proposed gating layer to other emerging deep learning techniques. Once the node in the gating layer has a resulting activation equal to zero, the information does not propagate from that node (i.e., it turns off the corresponding node in the regular hidden layer). This feature of the proposed model is similar to drop nodes as done in dropout [57]. However, the approaches have different effects. Removing the nodes in dropout is done at random, which can be interpreted as averaging the results of an ensemble of DNNs. In the gating layer, the nodes are intentionally removed by learning the relevance of the information propagated through the nodes. Instead of improving generalization, the gating layer aims to filter noisy information not related to the task.

The gating layer is similar to the attention model [58]. The attention model follows Equations 10 and 11. $\mathbf{c}_t$ is the embedding value fed into the higher layer at time t. $\mathbf{h}_n$ is a sequence of hidden values extracted from lower recurrent layer along the time axis, whose length is n. λ_{t_n} is the corresponding weight for the hidden value at each time step. The embedding is achieved with a weighted summation of the hidden values across time. By learning the weights, the attention model captures the importance of the outputs of the hidden layers at different time steps (temporal weighting in the network). In contrast, the current version of GNN does not have a recurrent layer in the model. The GNN learns the importance of the hidden layers from spatial information rather than temporal information. Its goal is to prevent information from propagating in the network, which is achieved with the gate $g()$. Since this function is a sigmoid, the resulting activity is close to either one or zero, functioning as a switch (spatial filtering in the network).

$$\mathbf{c}_t = \sum_n \lambda_{t_n}\mathbf{h}_n \qquad (10)$$

$$\sum_n \lambda_{t_n} = 1 \qquad (11)$$

Another similar technique is the highway connection [59]. However, the highway connection has a different purpose than the gating layer. Highway connections are utilized in very deep networks, allowing information to skip layers. The gating layer is mainly used to filter noisy or non-informative inputs. The gating layer can be applied when the input feature vector has uncertain or redundant information.

V. Experiments and Analysis

The evaluation uses the CRSS-4ENGLISH-14 corpus. The video features in this section correspond to the set described in Section III-C (i.e., the random features are only used in Section IV). The video frames are up-sampled using linear interpolation, matching the frame per second of the audio modality (e.g., 100 fps). Features with larger values can dominate the training of the deep learning structures. This is a problem for AV-ASR, since visual features can have greater values than audio features. Therefore, we normalize each feature by removing its mean and dividing by its standard deviation before concatenating the audiovisual features. The baseline models have observation models built with GMMs and a DNN. We also compare the approach with a MS-HMM.

The audio collected with the close-talking microphone during the clean recordings is used to train the GMM-HMM framework, which is implemented with Kaldi [52]. We use forced alignment using this model to obtain initial phonetic labels and train the DNN and GNN-frameworks. These models are implemented with a PDNN [53], which are then combined with HMMs to capture the temporal dynamic information. We set the mini-batch size to 256, using *Adagrad* across all the deep learning training [54]. We initialize all the neural networks using random initialization following the approach proposed by Glorot and Bengio [60] (uniform initializer). We use tri-gram as our language model built exclusively with the transcriptions of the training set. We use the Kneser-Ney smoothing for the language model, which is the default option in Kaldi. We back-off to bi-gram and uni-gram when the evidence for higher-order n-gram is not sufficient. The perplexity of our language model is 8.59. Notice that all the models use the same language model so the comparisons are fair. While the MS-HMM framework is implemented with HTK, we follow the same training procedure used in Kaldi.

We evaluate whether the differences in performance are statistically significant with the McNemar's test using the *speech recognition scoring toolkit* (SCTK) [61]. This test is commonly used to compare WERs. We assert performance at p-value=0.01.

A. Defining the GNN Structure

For the GMM-HMM system, we use the same setup described in Section IV-A. For the DNN-HMM system, we incrementally added hidden layers starting from three and going to six, each with 1,024 nodes. Our preliminary evaluation, not reported in this study, showed that four layers provide the best performance on the validation set. Therefore, we fixed the number of layers to four. For the proposed GNN, we use one gating layer and four regular hidden layers, each of them with 1,024 nodes. Notice that the GNN framework has roughly the same number of parameters as a DNN implemented with an additional layer. However, Section V-D shows that adding an extra layer to the DNN framework does not increase its performance so the comparisons are fair.

An important question in the GNN structure is the location of the gating layer that maximizes the performance of the system. In Figure 5(a), the gating layer affects the second hidden layer. We implement the proposed framework adding the gating layers at different levels. Table IV shows the WER on the validation set. This experiment was done with audiovisual data collected with the close-talking microphone and HD camera under clean recordings. The performance at the 4_{th} layer is significantly worse than the ones at other positions (p-value<0.01). The differences between other cases are not statistically significant. Since the performance at the 2_{nd} position is slightly better than other positions, we implement the gating layer on top of the second hidden layer. The first two hidden layers process the input features. After propagating the information to upper layers, the GNN will filter the noisy data. The last two fully connected hidden layers combine the remaining information, providing discriminative representation to the HMM framework. Based on this result, we implement the gating layer on top of the second hidden layer in the remaining experiments.

TABLE IV: Performance of the GNN on the validation set when we implement the gating layers at different levels. The "POSITION" column indicates the layer over which the gating layer is implemented.

POSITION	WER [%]
Input	3.79
1^{st}	3.74
2^{nd}	**3.69**
3^{nd}	3.73
4^{th}	4.67

TABLE V: Evaluation of individual modules with clean speech over the test set. The results are presented with and without feature normalization.

MODEL	NORMALIZATION	WER [%]
Audio GMM-HMM	N	4.62
Audio DNN-HMM	N	**3.70**
Audio DNN-HMM	Y	3.88
Audio GNN-HMM	Y	4.00
Audiovisual DNN-HMM	Y	4.20
Audiovisual GNN-HMM	Y	**3.70**

B. Contribution of Individual Modules

This section evaluates the performance of the proposed system by adding the building blocks of the system step-by-step to assess their contributions. This evaluation only uses the clean recordings for the train and test sets using the close-talking microphone and HD camera. Table V reports the results. The first row shows the performance of the GMM-HMM using acoustic features, without feature normalization, which achieves 4.62% WER. When we replace GMM with DNN, the performance significantly improves to 3.70% (p-value<0.01), showing the benefits of using a DNN over GMMs as the observation model. The third row in the table shows a drop in performance when we add feature normalization, obtaining a 3.88% WER. While this feature normalization is not needed for acoustic features, which tend to have similar ranges, it is important for visual features. The performance slightly drops when we add the gating layer (4.00% for the GNN-HMM framework, p-value<0.01). We observe an important result when we add the visual modality. For the DNN-HMM framework, adding visual features significantly reduces the performance of the system from 3.88% WER (audio only) to 4.20% WER (audiovisual) (p-value<0.01). However, the proposed GNN-HMM framework is able to significantly reduce the WER from 4.00% (audio only) to 3.70% (audiovisual) (p-value<0.01). These results with clean recordings illustrate the need of an audiovisual system that improves performance not only with noisy recordings, as commonly demonstrated in related studies, but also with clean recordings. The proposed GNN framework is able to achieve this goal.

C. Evaluation with Matched and Mismatched Conditions

This section presents rigorous evaluations for noisy and clean conditions for MS-HMM, GMM, DNN and GNN frameworks. The results shown in Section V-B do not have mismatch conditions, since the test and train sets were collected under the same conditions. In practice, a system will need to be effective even when it is tested with different microphones or cameras, or with different levels of noise. This section evaluates the robustness of the proposed approach with matched and mismatched conditions. For training, we use the data from the close-talking microphone and HD camera. For testing, we evaluate two conditions by using the data from the tablet, the close-talking microphone and the HD camera. The *channel matched condition* corresponds to audio from the close-talking microphone and video from the HD camera (i.e., train and test data recorded with the same microphones and camera). This condition includes the data streams with the best quality. The second condition uses audiovisual data collected by the tablet, and is referred to as *channel mismatched condition* (train and test data do not come from the same microphones and cameras). The channel mismatched condition resembles real-world video conference applications where the audio and video are collected about two meters away from the user using internal sensors of a portable device.

Table VI shows the results for the GMM, DNN and GNN systems, evaluated with A-ASR (rows one to three), V-ASR (rows four to six) and AV-ASR (rows eight to ten). It also includes the results of MS-HMM for audiovisual fusion (row seven). The first two columns correspond to the channel matched conditions, and the last two columns correspond to the channel mismatched conditions. For each of these conditions, we provide the results on the test set with clean and noisy audio recordings. Notice that the noisy recordings are collected at the end of the session using exactly the same prompted speech used for the clean conditions (Sec. III-A). For each condition, we mark with an asterisk when one system is significantly better than other systems. For example, the proposed GNN-HMM framework achieves the best performance for the AV-ASR task, with channel matched condition, and with clean speech (3.70% WER).

The results on Table VI show that the audiovisual GNN-HMM framework has the best performance across all conditions (last row of Table VI). The WER is lower than all the A-ASR systems. This result shows the strengths of the proposed approach to deal with concatenated features. When we only consider systems trained with either audio or visual features, we consistently observe that DNN is the best method. The strength of the GNN framework is when the system is trained by combining these modalities. Notice that the systems trained with visual features have lower performance than the systems trained with audio features, as expected. Audiovisual systems trained with GMMs or a DNN only improve the performance of audio-only frameworks when the audio is noisy. For example, the WER for the DNN decreases from 33.43% (audio-only) to 15.88% (audiovisual) for the channel mismatched condition under noisy recordings (p-value<0.01). For clean speech, the WER of these systems increases after adding visual features. This problem is not observed in the GNN framework, where the WER decreases across all conditions. It is interesting to observe that GNN preserves the complementary information of visual features, which are less discriminative than acoustic features, improving the performance even for clean speech under channel matched conditions.

TABLE VI: Performance of the proposed GNN framework with the baseline methods for A-ASR, V-ASR and AV-ASR over the test set. The results are presented for channel matched conditions and channel mismatched conditions using clean and noisy recordings ($*$ indicates one method is significantly better than others for a given condition).

MODEL	Channel Matched		Channel Mismatched	
	Clean WER [%]	Noise WER [%]	Clean WER [%]	Noise WER [%]
Audio GMM-HMM	4.62	7.02	4.91	39.77
Audio DNN-HMM	**3.70**	6.09*	4.64	33.43*
Audio GNN-HMM	4.00	7.52	5.29	38.68
Visual GMM-HMM	66.56	69.52	75.05	76.72
Visual DNN-HMM	64.52*	65.11*	70.66*	70.47*
Visual GNN-HMM	71.53	72.52	76.41	76.65
Audiovisual MS-HMM	14.03	14.51	18.79	22.34
Audiovisual GMM-HMM	23.27	24.19	24.66	30.70
Audiovisual DNN-HMM	4.20	4.87	15.48	15.88
Audiovisual GNN-HMM	**3.70***	**4.32***	**4.05***	**11.48***

When we test the systems with clean data, the proposed AV-GNN-HMM approach can maintain or even improve the performance of the audio only systems. This is not observed with MS-HMM, where the performance is significantly lower than A-ASR systems (p-value<0.01). For the channel matched condition, the performance in the AV-GNN-HMM system is preserved even when the WER of the visual GNN is as high as 71.53%. The results are even more impressive for the channel mismatched condition, when the WER decreases from 4.64% to 4.05% (p-value<0.01). The system is robust to the microphone mismatch where we train with a close-talking microphone and test with the tablet microphone located about two meters from the user (distant speech). The results indicate that the gating mechanism is able to filter out confusing information in the input.

When we test the systems with noisy speech, the WER of the audio-only systems decreases, especially for the channel mismatched conditions where the microphone is closer to the audio speaker playing the noise. As shown in Figure 2, the SNR for the noisy recordings in the channel mismatched condition is lower, which explains this result. For example, the performance of the audio-only DNN system dropped from 3.70% to 6.09% for the channel matched condition (p-value<0.01), and from 4.64% to 33.43% for the channel mismatched condition (p-value<0.01). The performance for the visual-only systems for the noisy conditions is similar to the clean conditions, since the noise only affects the audio. We do not observe a drop in performance associated with changes in articulations due to Lombard speech. However, we observe a drop in performance due to camera mismatch when we train with visual features extracted from the HD camera and test with features from the tablet camera (results from the channel

TABLE VII: Comparison of the proposed framework with DNN implemented with four and five hidden layers. The network parameters are shown in million ("M" for short). The evaluation considers the best and worst testing scenarios from Table VI: channel matched condition with clean recordings, channel mismatched condition with noisy recordings ($*$ indicates one method is significantly better than others for a given condition).

MODEL	PARA # [M]	Channel Matched	Channel Mismatched
		Clean, WER [%]	Noise, WER [%]
DNN (4 layers)	$\sim$ 7.6	4.20	15.88
DNN (5 layers)	$\sim$ 8.6	4.33	17.11
GNN	$\sim$ 8.6	**3.70$*$**	**11.48$*$**

mismatched condition). The proposed GNN system is able to improve the performance under noisy conditions when we consider audiovisual features. By adding visual cues, the GNN achieves a 11.48% WER, which represents a 27.20% (absolute) improvement over the audio-only system (p-value<0.01). The evaluation showed the proposed AV-GNN system is robust against different mismatches, providing evidences that this framework is suitable for practical applications.

D. Number of Parameters

The audiovisual GNN framework has about 8.6 million parameters (0.71M input layer + four layers × 1.05M + 2.68M output layer + 1.05M GNN layer). The number of parameters is roughly the same as the DNN implemented with five hidden layers (0.71M input layer + 5 layers × 1.05M +2.68M output layer). As mentioned in Section V-A, the DNNs are implemented with four regular layers, which have 7.6M parameters (i.e., about 1M less parameters). It may be argued that a reason for GNN to outperform regular DNN is the additional parameters. Therefore, this section evaluates a regular DNN implemented with four and five layers, comparing the performance with the proposed GNN framework. Table VI shows that the best performance was observed with close-talking microphone and HD camera with clean speech. The worst performance was observed with the tablet sensors under noise condition. Therefore, we only consider these two extreme cases. The systems are consistently trained with data from clean recordings collected with close-talking microphone and HD camera.

Table VII shows the results. Adding an extra layer to the DNN framework decreases its performance in both conditions. The results show that (1) the comparisons in previous sections are fair, (2) the GNN approach outperforms the DNN for all the configurations.

VI. Conclusions

This study introduced the gating layer which is ideal for dealing with multimodal deep learning. During training, the gating layer learns to remove noisy or redundant information. We evaluated the framework to build a large vocabulary audiovisual speech recognition system. Common audiovisual approaches only provide benefits when the audio is noisy. For clean speech, adding visual features tends to reduce the performance. The proposed system addresses these problems with the gating layer, achieving improved performance even with clean speech. The benefits of the proposed approach was demonstrated with extensive evaluations using the CRSS-4ENGLISH-14 corpus, which is one of the largest audiovisual corpus in the community. The proposed approach outperformed the conventional GMM approach and the state-of-the-art DNN approach for all the experimental evaluations. It also achieved better performance than MS-HMM. Unlike alternative frameworks, the proposed approach can maintain performance when the audio is clean. When tested with noisy data, the approach showed improved robustness compared with GMM and DNN approaches. This framework opens realistic opportunities to introduce AV-ASR systems in real-world applications.

The evaluation showed that by adding 25% more parameters (one gating layer), the proposed GNN framework can obtain good performance using concatenated feature as input. There are several research directions motivated by this study. In several areas, the input feature vector may introduce redundant information into the system. The gating layer offers a principled framework to deal with this problem. The use of this framework in other domains is left as future work. Currently, we only evaluate one gating layer. However, adding gating layers at multiple levels can improve the capability of the system to capture useful information. The evaluation considered models trained with clean speech. An extension of the approach is to explore training schemes with noisy signals, which we expect will improve the performance of the system. The experimental evaluation only considered noisy acoustic features. Visual features can also be "noisy" due to occlusion, illumination or blurred images. We will evaluate these cases in our future work. Furthermore, we will explore other visual features which may provide better performance, including extracting features with CNNs from raw images. Finally, we will compare the gating layers with systems that dynamically change the weights of the modalities, setting values according to the reliability of their features. These systems provide an alternative solution to this problem.

Acknowledgment

This study was funded by the National Science Foundation (NSF) CAREER grant IIS-1453781.

References

[1] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, November 2012.

[2] G. Saon, G. Kurata, T. Sercu, K. Audhkhasi, S. Thomas, D. Dimitriadis, X. Cui, B. Ramabhadran, M. Picheny, L.-L. Lim, B. Roomi, and P. Hall, "English conversational telephone speech recognition by humans and machines," *ArXiv e-prints (arXiv:1703.02136)*, March 2017.

[3] L. Brancazio and J. Miller, "Use of visual information in speech perception: Evidence for a visual rate effect both with and without a McGurk effect," *Attention, Perception, & Psychophysics*, vol. 67, no. 5, pp. 759–769, July 2005.

[4] K. Helfer, "Auditory and auditory-visual perception of clear and conversational speech," *Journal of Speech, Language, and Hearing Research*, vol. 40, pp. 432–443, April 1997.

[5] A. Macleod and Q. Summerfield, "A procedure for measuring auditory and audiovisual speech-reception thresholds for sentences in noise: Rationale, evaluation, and recommendations for use," *Journal British Journal of Audiology*, vol. 24, no. 1, pp. 29–43, 1990.

[6] C. Neti, G. Potamianos, J. Luettin, I. Matthews, H. Glotin, D. Vergyri, J. Sison, A. Mashari, and J. Zhou, "Audio-visual speech recognition," Workshop 2000 Final Report, Technical Report 764, October 2000.

[7] G. Potamianos, C. Neti, G. Gravier, A. Garg, and A. Senior, "Recent advances in the automatic recognition of audiovisual speech," *Proceedings of the IEEE*, vol. 91, no. 9, pp. 1306–1326, September 2003.

[8] J. Luettin, G. Potamianos, and C. Neti, "Asynchronous stream modeling for large vocabulary audio-visual speech recognition," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2001)*, Salt Lake City, UT, USA, May 2001, pp. 169–172.

[9] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Ng, "Multimodal deep learning," in *International conference on machine learning (ICML2011)*, Bellevue, WA, USA, June-July 2011, pp. 689–696.

[10] D. Stewart, R. Seymour, A. Pass, and J. Ming, "Robust audio-visual speech recognition under noisy audio-video conditions," *IEEE Transactions on Cybernetics*, vol. 44, no. 2, pp. 175–184, February 2014.

[11] J. Huang and B. Kingsbury, "Audio-visual deep learning for noise robust speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2013)*, Vancouver, BC, Canada, May 2013, pp. 7596–7599.

[12] K. Noda, Y. Yamaguchi, K. Nakadai, H. Okuno, and T. Ogata, "Audio-visual speech recognition using deep learning," *Applied Intelligence*, vol. 42, no. 4, pp. 722–737, June 2015.

[13] K. Thangthai, R. Harvey, S. Cox, and B. Theobald, "Improving lip-reading performance for robust audiovisual speech recognition using DNNs." in *Joint Conference on Facial Analysis, Animation and Auditory-Visual Speech Processing (FAASP)*, Vienna, Austria, September 2015, pp. 127–131.

[14] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, November 1997.

[15] F. Tao and C. Busso, "Lipreading approach for isolated digits recognition under whisper and neutral speech," in *Interspeech 2014*, Singapore, September 2014, pp. 1154–1158.

[16] X. Zhang, R. Mersereau, and M. Clements, "Audio-visual speech recognition by speechreading," in *International Conference Digital Signal Processing (DSP 2002)*, vol. 2, Santorini, Greece, July 2002, pp. 1069–1072.

[17] X. Zhang, C. Broun, R. Mersereau, and M. Clements, "Automatic speechreading with applications to human-computer interfaces," *EURASIP Journal on Advances in Signal Processing*, vol. 1, pp. 1228–1247, January 2002.

[18] G. Potamianos, C. Neti, J. Luettin, and I. Matthews, "Audiovisual automatic speech recognition," in *Audio-Visual Speech Processing*, G. Bailly, P. Perrier, and E. Vatikiotis-Bateson, Eds. Cambridge, UK: Cambridge University Press, February 2015, pp. 193–247.

[19] E. Benhaim, H. Sahbi, and G. Vitte, "Designing relevant features for visual speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2013)*, Vancouver, BC, Canada, May 2013, pp. 2420–2424.

[20] A. Shaikh, D. Kumar, W. Yau, M. Che Azemin, and J. Gubbi, "Lip reading using optical flow and support vector machines," in *International Congress on Image and Signal Processing (CISP 2010)*, Yantai, China, October 2010, pp. 327–330.

[21] T. Stafylakis and G. Tzimiropoulos, "Combining residual networks with LSTMs for lipreading," in *Interspeech 2017*, Stockholm, Sweden, August 2017, pp. 3652–3656.

[22] J. Chung, A. Senior, O. Vinyals, and A. Zisserman, "Lip reading sentences in the wild," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017)*, Honolulu, HI, USA, July 2017, pp. 3444–3453.

[23] S. Petridis, Z. Li, and M. Pantic, "End-to-end visual speech recognition with LSTMs," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2017)*, New Orleans, LA, USA, March 2017, pp. 2592–2596.

[24] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE,*, vol. 86, no. 11, pp. 2278–2324, November 1998.

[25] K. Noda, Y. Yamaguchi, K. Nakadai, H. Okuno, and T. Ogata, "Lipreading using convolutional neural network," in *Interspeech 2014*, Singapore, September 2014, pp. 1149–1153.

[26] S. Petridis and M. Pantic, "Deep complementary bottleneck features for visual speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2016)*, Shanghai, China, March 2016, pp. 2304–2308.

[27] S. Tamura, H. Ninomiya, N. Kitaoka, S. Osuga, Y. Iribe, K. Takeda, and S. Hayamizu, "Audio-visual speech recognition using deep bottleneck features and high-performance lipreading," in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA 2015)*, Hong Kong, December 2015, pp. 575–582.

[28] C. Bregler, H. Hild, S. Manke, and A. Waibel, "Improving connected letter recognition by lipreading," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 1993)*, vol. 1, Minneapolis, MN, USA, April 1993, pp. 557–560.

[29] Y. Mroueh, E. Marcheret, and V. Goel, "Deep multimodal learning for audio-visual speech recognition," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2015)*, Brisbane, Australia, April 2015, pp. 2130–2134.

[30] G. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 30–42, January 2012.

[31] A. Mohamed, G. Dahl, and G. Hinton, "Acoustic modeling using deep belief networks," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 14–22, January 2012.

[32] G. Hinton, S. Osindero, and Y. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, July 2006.

[33] A. K. Katsaggelos, S. Bahaadini, and R. Molina, "Audiovisual fusion: Challenges and new approaches," *Proceedings of the IEEE*, vol. 103, no. 9, pp. 1635–1653, September 2015.

[34] H. Ninomiya, N. Kitaoka, S. Tamura, and Y. I. K. Takeda, "Integration of deep bottleneck features for audio-visual speech recognition," in *Interspeech 2015*, Dresden, Germany, September 2015, pp. 563–567.

[35] S. Gergen, S. Zeiler, A. Abdelaziz, R. Nickel, and D. Kolossa, "Dynamic stream weighting for turbo-decodingbased audiovisual ASR," in *Interspeech 2016*, San Francisco, CA, USA, September 2016, pp. 2135–2139.

[36] H. Meutzner, N. Ma, R. Nickel, C. Schymura, and D. Kolossa, "Improving audio-visual speech recognition using deep neural networks with dynamic stream reliability estimates," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2017)*, New Orleans, LA, USA, March 2017, pp. 5320–5324.

[37] J. Chung and A. Zisserman, "Out of time: automated lip sync in the wild," in *Asian Conference on Computer Vision (ACCV 2016 Workshop)*, ser. Lecture Notes in Computer Science, C. Chen, J. Lu, and K. Ma, Eds. Taipei, Taiwan: Springer Berlin Heidelberg, November 2016, vol. 10117, pp. 251–263.

[38] E. Patterson, S. Gurbuz, Z. Tufekci, and J. Gowdy, "CUAVE: A new audio-visual database for multimodal human-computer interface research," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2002)*, Orlando, FL, USA, May 2002, pp. 2017–2020.

[39] M. Cooke, J. Barker, S. Cunningham, and X. Shao, "An audio-visual corpus for speech perception and automatic speech recognition," *The Journal of the Acoustical Society of America*, vol. 120, no. 5, pp. 2421–2424, November 2006.

[40] I. Anina, Z. Zhou, G. Zhao, and M. Pietikäinen, "OuluVS2: a multi-view audiovisual database for non-rigid mouth motion analysis," in *IEEE International Conference on Automatic Face and Gesture Recognition (FG 2015)*, Ljubljana, Slovenia, May 2015, pp. 1–5.

[41] N. Harte and E. Gillen, "TCD-TIMIT: An audio-visual corpus of continuous speech," *IEEE Transactions on Multimedia*, vol. 17, no. 5, pp. 603–615, May 2015.

[42] D. Burnham, D. Estival, S. Fazio, J. Viethen, F. Cox, R. Dale, S. Cassidy, J. Epps, R. Togneri, M. Wagner, Y. Kinoshita, R. Göcke, J. Arciuli, M. Onslow, T. Lewis, A. Butcher, and J. Hajek, "Building an audio-visual corpus of Australian English: Large corpus collection with an economical portable and replicable black box," in *Interspeech 2011*, Florence, Italy, August 2011, pp. 841–844.

[43] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal, G. Lathoud, M. Lincoln, A. Lisowska, I. McCowan, W. Post, D. Reidsma, and P. Wellner, "The AMI meeting corpus: A pre-announcement," in *Machine Learning for Multimodal Interaction (MLMI 2005)*, ser. Lecture Notes in Computer Science, R. S. and B. S., Eds. Edinburgh, UK: Springer Berlin Heidelberg, July 2005, pp. 12–21.

[44] V. M. Stanford, "NIST speech SNR tool," https://www.nist.gov/information-technology-laboratory/iad/mig/nist-speech-signal-noise-ratio-measurements, December 2005.

[45] X. Xiong and F. D. la Torre, "Supervised descent method and its applications to face alignment," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2013)*, Portland, OR, USA, June 2013, pp. 532–539.

[46] F. Tao, J. Hansen, and C. Busso, "An unsupervised visual-only voice activity detection approach using temporal orofacial features," in *Interspeech 2015*, Dresden, Germany, September 2015, pp. 2302–2306.

[47] F. Tao, J. L. Hansen, and C. Busso, "Improving boundary estimation in audiovisual speech activity detection using Bayesian information criterion," in *Interspeech 2016*, San Francisco, CA, USA, September 2016, pp. 2130–2134.

[48] F. Tao and C. Busso, "Bimodal recurrent neural network for audiovisual voice activity detection," in *Interspeech 2017*, Stockholm, Sweden, August 2017, pp. 1938–1942.

[49] G. Gravier, G. Potamianos, and C. Neti, "Asynchrony modeling for audio-visual speech recognition," in *Proc. Human Language Technology Conference*, San Diego, CA, 2002.

[50] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK Book*. Entropic Cambridge Research Laboratory, Cambridge, England, December 2006.

[51] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, Feb 1989.

[52] D. Povey, A. Ghosha, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, and J. Silovsky, "The Kaldi speech recognition toolkit," in *IEEE Workshop on Automatic Speech Recognition & Understanding (ASRU)*, Waikoloa, HI, USA, December 2011.

[53] Y. Miao, "Kaldi+ PDNN: building DNN-based ASR systems with Kaldi and PDNN," *ArXiv e-prints (arXiv:1401.6984)*, January 2014.

[54] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, vol. 12, pp. 2121–2159, July 2011.

[55] D. Rumelhart, G. Hinton, and R. Williams, "Learning representations by back-propagating errors," in *Cognitive modeling*, T. Polk and C. Seifert, Eds. Cambridge, MA, USA: A MIT Press, August 2002, pp. 213–220.

[56] I. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio, "Maxout networks," in *International Conference on Machine Learning (ICML 2013)*, Atlanta, GA, USA, June 2013, pp. 1–9.

[57] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, June 2014.

[58] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, "End-to-end attention-based large vocabulary speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2016)*, Shanghai, China, March 2016, pp. 4945–4949.

[59] R. Srivastava, K. Greff, and J. Schmidhuber, "Training very deep networks," in *Advances in neural information processing systems (NIPS2015)*, Montreal, Canada, December 2015, pp. 2377–2385.

[60] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *International Conference on Artificial Intelligence and Statistics (AISTATS 2010)*, Sardinia, Italy, May 2010, pp. 249–256.

[61] "NIST speech recognition scoring toolkit," https://www.nist.gov/itl/iad/mig/tools, October 2015, accessed Nov., 2017.