ULTRASOUND BASED GESTURE RECOGNITION

Amit Das[*]

Dept. of Electrical and Computer Engineering
University of Illinois, IL, USA
amitdas@illinois.edu

Ivan Tashev, Shoaib Mohammed

Microsoft Research
One Microsoft Way, Redmond, WA, USA
{ivantash, shoaib}@microsoft.com

ABSTRACT

In this study, we explore the possibility of recognizing hand gestures using ultrasonic depth imaging. The ultrasonic device consists of a single piezoelectric transducer and an 8 - element microphone array. Using carefully designed transmit pulse, and a combination of beamforming, matched filtering, and cross-correlation methods, we construct ultrasound images with depth and intensity pixels. Thereafter, we use a combined Convolutional (CNN) and Long Short-Term Memory (LSTM) network to recognize gestures from the ultrasound images. We report gesture recognition accuracies in the range 64.5-96.9%, based on the number of gestures to be recognized, and show that ultrasound sensors have the potential to become low power, low computation, and low cost alternatives to existing optical sensors.

Index Terms— gesture recognition, ultrasound depth imaging, beamforming, convolutional neural networks, long short-term memory

1. INTRODUCTION

Mobile, interactive devices are emerging as the next-frontier of personalized computing. Providing effective input-output (IO) modalities - gestures, touch, voice, *etc.* - is a key challenge for such devices [1], [2]. Today, hand-gesture based IO devices are broadly enabled by optical sensing [3]. They rely on estimating distances to target objects by measuring the time-of-flight (ToF) in air. ToF is the duration between the time a probe signal is transmitted to the target object and the time the reflected version of the probe signal is received. It is measured as $\frac{2d}{c}$, where d is the distance of the target object and c = 343 m/s is the speed of sound in air. Unfortunately, optical sensors face high-energy costs because of illumination overhead and processing complexities (capture, synchronization and analysis). This limits their use in mobile, interactive devices like head-mounted-displays (HMD) and wearables, where energy costs carry a big premium. For instance, consider an HMD running on a 1500 mAH (3.8 V) battery with an IO energy-budget of 20% (*i.e.*, 4104 J). Assuming that an optical sensor consumes 2.5 W of power, the HMD can barely support a total of 500 gestures with each gesture lasting 3 seconds (IO budget/energy-per-gesture = 4104 J/7.5 J). Power limitations like these thus raise the need for alternative technologies that can be utilized to recognize gestures with low energy. One such alternative that we explore in this paper is ultrasound imaging. Our choice is motivated by the fact that ultrasound sensors require only a fraction of the power consumed by optical sensors. Going back to our example of the HMD, if we were to use an ultrasonic sensor ($\leq$15 mW) instead of an optical sensor, the device would be able to support nearly 100k gestures within the same energy budget; a compelling 200 fold increase.

2. PRIOR WORK

Several interesting approaches exist in optical sensing and to a limited degree in ultrasonic sensing. For instance, in [4], the authors capture depth images of static hand poses and classify them using a 3D nearest-neighbor classifier; and in [5], the authors use depth images in conjunction with a probabilistic state-space temporal model to track fast-moving objects.

In [6], the authors use doppler spectra of ultrasound signals together with a GMM-based classifier to distinguish humain gait. In a follow-up work, they extend this idea to recognize static hand gestures [7]. In [8], the authors augment the acoustic signals with ultrasound doppler signals for multimodal speech recognition. They note that ultrasound signals can potentially add valuable information to the acoustic signals, especially in noisy environments. In [9], the authors use an 8-element loudspeaker array and sound-source localization (SSL) to create acoustic depth maps of static poses positioned 3 m away. Our proposed system is related to [9] but applies to a different setting (recognize dynamic hand-gestures up to 1 m away), which precludes the use of complex SSL algorithms like MUSIC (multiple signal classification). Thus, our work extends [9] in the following ways:

A. To insonify images, we use only one loudspeaker instead of 8 (7$\times$ power savings).

B. We use one-shot acquisition to capture the entire image. This allows us to achieve real-time sensing (rates up to 170 frames-per-second (fps)) necessary to recognize fast moving gestures. This is in contrast to [9], where one shot per transducer was needed (limiting sensing rate in similar scenarios to 20 fps).

C. We propose a new dual-input CNN-LSTM network that outputs a single hand gesture for a given sequence of ultrasonic images as input.

[*]Author performed the work while at Microsoft Research, Redmond, WA.

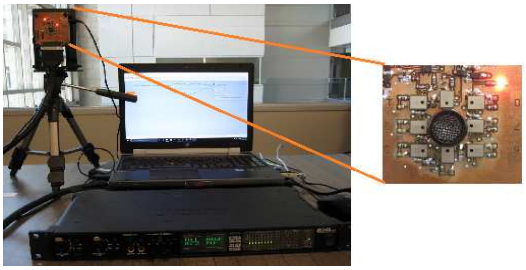

Fig. 1: Left: Hardware Set-Up; Right: Ultrasonic piezoelectric transducer at the center and an 8-element microphone array around it in a circular configuration.

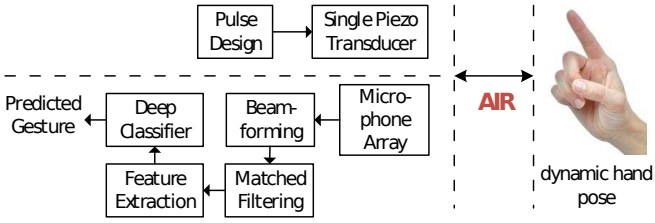

Fig. 2: Block Diagram of the proposed approach.

The rest of the paper is organized as follows. In the next section, we describe the proposed system including the various sub-components involved. In Section 4, we provide measurement results and conclude in Section 5.

3. SYSTEM APPROACH

Fig. 1 shows our hardware setup. It consists of one piezoelectric transducer placed at the center of an 8-element circular array of MEMS microphones, an audio interface (digital-to-analog and analog-to-digital converter), and a laptop for controlling the signals. The transducer emits ultrasound pulses in the 36-44 kHz range. A block diagram of the system is shown in Fig. 2. Next, we describe various components shown in the block diagram: pulse design, beamforming, matched filtering, feature extraction, and recognition using CNN-LSTM.

3.1. Pulse Design

The transmit pulse requirements are as follows: (a) its autocorrelation should have one sharp peak for easier detection of echoes using the cross-correlation method, (b) since the piezoelectric transducer resonates around 40 kHz, the transmit pulse should be band limited to 36-44 kHz, (c) the pulse is also time limited since the width of the pulse T_{P} should be smaller than the minimal time-of-flight (ToF_{min}); for d_{min} = 30 cm, ToF_{min} = 1.7 ms. To meet these constraints, we use a linear frequency modulated (LFM) chirp of duration T_{P} = 1.5 ms and bandlimited to 36-44 kHz. The amount of spectral leakage of the LFM chirp is inversely proportional to the duration of the chirp. We therefore apply a rectangular filter in the frequency domain in the desired frequency range (36-44 kHz) followed by a Hamming window in the time domain to reduce the spreading (correlations) in the autocorrelation function.

3.2. Beamforming and Matched Filtering

The ultrasonic signals, sampled at 192 kHz, are received by an M element microphone array (here $M = 8$) and combined to form a single received signal. We use the Minimum Variance Distortionless Response (MVDR) beamformer (BF) [10] following the overall beamformer architecture as described in [11]. Let $S(f,\psi)$ be the target source located in some direction $\psi = (\phi,\theta)$ (where ϕ = azimuth, θ = elevation) and emitting frequency f. Let $\mathbf{D}(f,\psi)$ be the $M\times 1$ capture vector of the microphone array in the look direction ψ. Let $\mathbf{N}(f)$ be the $M\times 1$ noise vector of the microphone array at frequency f. The BF applies M weights to the received signal to form a composite signal $Y(f)$ where,

$$Y(f) = \mathbf{W}^T(f,\psi)\mathbf{D}(f,\psi)S(f,\psi) + \mathbf{W}^T(f,\psi)\mathbf{N}(f). \quad (1)$$

The objective of MVDR BF is to design the weights $\mathbf{W}(f,\psi)$ such that the noise power is minimized while keeping the target signal undistorted. Solving this constrained optimization problem results in the optimal weights given by,

$$\mathbf{W}(f,\psi) = \frac{\mathbf{D}^{\mathrm{H}}(f,\psi)\mathbf{C}_{\mathrm{NN}}^{-1}}{\mathbf{D}^{\mathrm{H}}(f,\psi)\mathbf{C}_{\mathrm{NN}}^{-1}\mathbf{D}(f,\psi)}, \quad (2)$$

where $\mathbf{C}_{\mathrm{NN}}^{-1}$ is the $M\times M$ inverse noise covariance matrix of the microphone array. The elements of $\mathbf{C}_{\mathrm{NN}}^{-1}$ were computed *apriori* in a room similar to the operating environment. Since $\mathbf{C}_{\mathrm{NN}}^{-1}$ is not updated, our beamformer is time-invariant and can be designed offline with [12]. During real-time operation, only an inner-product of the weight vector with the received signal is required to compute the BF signal.

The field of view (FoV) was limited to the range ±40° horizontally (azimuth) and vertically (elevation). Based on the beamwidth, we set the beams every 5°. This yields 17×17 beams, and thus, the total number of look directions (pixels) to construct a single image is $17\times 17 = 289$. All angles were measured with respect to a reference point located at $(\phi_0,\theta_0) = (0°, 0°)$. The location of the piezoelectric transducer, which is also the center of the microphone array, was considered as the reference point (ϕ_0,θ_0).

After BF, we do matched filtering (MF) on the output of the BF since it is optimal in the sense that it maximizes the SNR of the received signal when corrupted by white noise. If $y(n)$ is the output of BF and $s(n)$ is the transmit pulse from Section 3.1, then the output of the MF is $x(n) = y(n) * s(-n)$.

3.3. Feature Extraction

We use two kinds of features: depth and intensity features. The depth ($d^\star$) is extracted by finding the peaks in the cross-correlation method as follows:

$$\begin{aligned} R_{XS}(\tau) &= \mathrm{FFT}^{-1}[X(f)S^\star(f)] \\ \tau^\star &= \underset{\tau\in[\mathrm{ToF_{min}},\mathrm{ToF_{max}}]}{\arg\max} R_{XS}(\tau) \\ d^\star &= \frac{c\tau^\star}{2} \end{aligned} \quad (3)$$

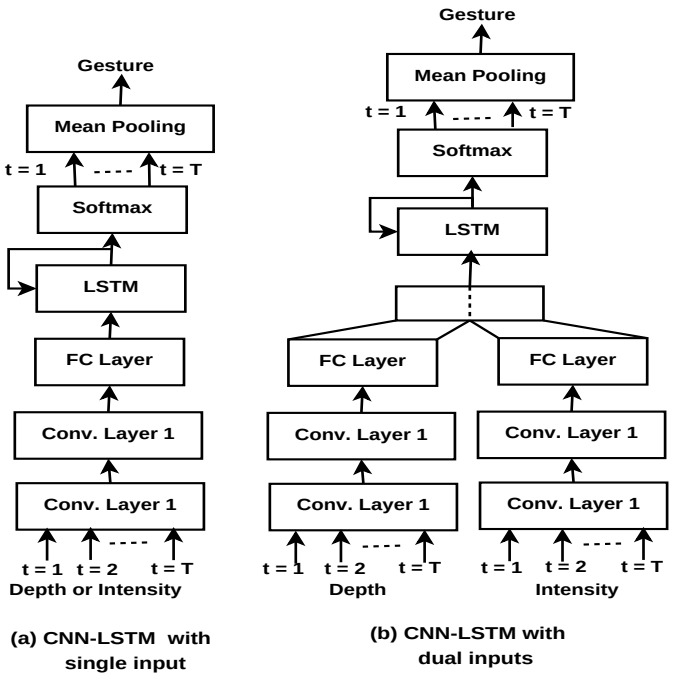

Fig. 3: CNN-LSTM architecture for gesture recognition

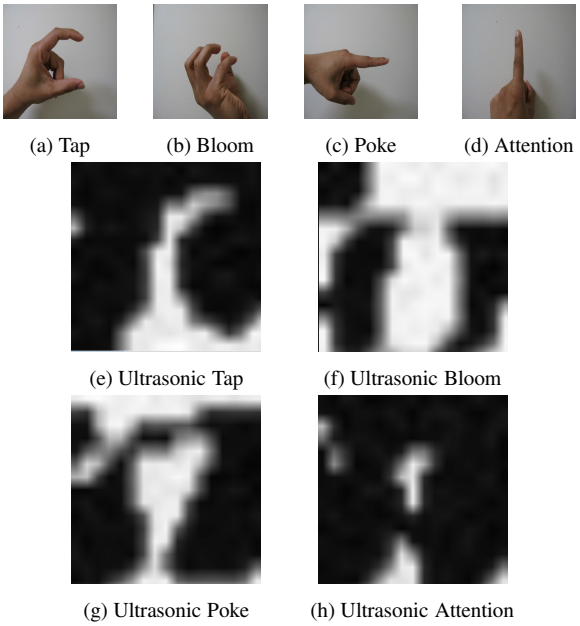

Fig. 4: Optical and ultrasonic images of different gestures

The intensity ($I^\star$) is simply the L_2 norm of the signal around $\tau^\star$, i.e., $I^\star = \int_{\tau^\star - \frac{T_P}{2}}^{\tau^\star + \frac{T_P}{2}} |x(t)|^2 dt$.

3.4. Recognition

The recognition stage is a sequence learning problem, where for an arbitrary length input sequence $\langle x_1, x_2, \cdots, x_T \rangle$ (the value of T depends on the length of the sequence), the objective is to produce a single label (or gesture) y summarizing the input sequence. In other words, the learning problem is to estimate the function f where $f : \langle x_1, x_2, \cdots, x_T \rangle \mapsto y$.

We use a combination of CNN and LSTM, since this is the state-of-the-art classifier and has shown to be useful for activity-recognition tasks [13] which evolve both in space and time. This is illustrated in Fig. 3(a). The input features to the CNN consists of either depth or intensity images. In this study, a single layer of CNN, which is referred to as convolution layer (CL), consists of three operations - convolution, rectification, and max pooling. First, the input image over a small region is convolved with a kernel (or convolution weights) to produce an activation local to that small region. By repeating the convolution operation using the same kernel over different local regions of the input image, it is possible to detect patterns captured by the kernels regardless of the absolute position of the pattern in the input image. Next, the activations undergo a non-linear transformation through a rectified linear unit (ReLU). Finally, dimension reduction of the activations is achieved by carrying out max pooling over non-overlapping regions. Our CNN architecture consists of two such CLs followed by a fully connected (FC) layer. The resulting high-level features generated by the CNN are better at preserving local invariance properties than the raw input features [14].

Although the CNN features capture depth in space, they do not capture depth in time. Since gestures evolve both in space and time, additional information about temporal dynamics can be captured by incorporating temporal recurrence connections using recurrent neural networks (RNNs). RNNs have been proven to be successful in speech recognition [15], speech enhancement [16, 17] and language modeling tasks [18]. However, they are difficult to train due to the vanishing/exploding gradients problem over long time steps [19]. LSTMs overcome this problem by incorporating memory cells that allow the network to learn to selectively update or forget previous hidden states given new inputs. The unidirectional left-to-right LSTM of [20] was used in this study. The high-level features of the CNN were input to the LSTM to capture the temporal structure of the gesture. Thus, temporal connections occur only at the LSTM block. For the final classification stage, the outputs of the LSTM were input to a softmax layer. All weights in the CNN-LSTM network are trained using supervised cross-entropy training. During testing, for every input image x_t at time step t, the CNN-LSTM network generates a posterior probability for gesture c, i.e., $p(\hat{y}_t = c|x_t), c \in C$ where C is the set of gestures. Since the objective is to generate a single gesture for the entire sequence from $t = 1$ to $t = T$, we simply do a mean pooling of the posteriors of all the gestures and pick the gesture with the highest mean posterior. To improve the accuracy further, we make use of both depth and intensity features since they can provide useful complementary information when used in conjunction. Thus, we propose the dual input CNN-LSTM architecture as shown in Fig. 3(b). The left CNN processes the depth features whereas the right CNN processes the in-

tensity features simultaneously. The outputs of the two CNNs are stacked together and fed as inputs to the LSTM.

4. EXPERIMENTS AND RESULTS

We selected five types of gestures in this study, *viz.* tap, bloom, poke, attention, and random gesture. The first four gestures have well-defined hand or finger movements. The fifth gesture (random) is any arbitrary gesture which is not similar to the other four well-defined gestures. These five gestures are grouped into six categories as follows:

- CAT 5: Tap, Bloom, Poke, Attention, Random
- CAT 4a: Tap, Bloom, Poke, Attention
- CAT 4b: Tap, Bloom, Attention, Random
- CAT 3a: Tap, Poke, Attention
- CAT 3b: Tap, Bloom, Attention
- CAT 2: Tap, Attention

A total of 40 subjects of ages between 20-60 years were asked to perform gestures within the FoV of the ultrasonic camera and within a distance of 30-100 cm from the device. Each subject was asked to perform the five gestures while repeating each type 20 times. Consequently, for 40 subjects, a total of 4000 gestures were collected. Out of these, gestures from 5 subjects were used as development set, and 4 others as test set. The remaining gestures were used for training. Each gesture was about 3-4 seconds long and captured at a rate of ~ 50 fps. Samples of 17×17 ultrasonic images of the gestures are shown in Figs. 4(e)-(h). Also shown are representative optical images for comparison in Figs. 4(a)-(d) (though not of the same instance as the ultrasonic gesture). The bright and dark regions of the ultrasonic images are indicative of the presence and absence of objects respectively. Fig. 4(e) shows an ultrasonic image of the tap gesture. A bent index finger on the upper half of the image and a partial thumb in the lower right corner is clearly visible. The three fingers and the spaces between them represent a bloom gesture in Fig. 4(f). Most of the cues about the poke gesture is present in the bright horizontal line in the upper half of Fig. 4(g). Similarly, the vertical bright line represents the vertical index finger of the attention gesture in Fig. 4(h).

Next, we present the results for the CNN-LSTM network of Fig. 3(a). The network was trained using CNTK [21]. Two different kinds of features were used for the CNN-LSTM - depth and intensity. For both features, the 2D kernel size was 2×2. The stride lengths for both the horizontal and vertical strides were 1. Zero-padding was used at the image edges. These settings were used for both the convolutional layers, CL1 and CL2. Max pooling was performed over small regions of size 2×2 with non-overlapping horizontal and vertical strides of lengths 2. The difference between the depth and intensity CNN-LSTMs is in the number of kernels in CL1 and CL2. We found that 16 and 32 kernels for CL1 and CL2 respectively were suitable for depth features. For intensity features, we found 16 kernels suitable for both CL1 and CL2. Additionally, we used a dropout factor of 0.2 to improve generalization. The output dimension of the FC layer was 128.

Feature	CAT 5	CAT 4a	CAT 4b	CAT 3a	CAT 3b	CAT 2
D	49.75	63.13	53.75	77.50	74.17	96.88
I	52.25	60.94	50.31	77.08	68.75	89.38
D+Ctx	52.25	67.81	60.00	74.58	76.25	96.88
I+Ctx	59.75	67.50	64.38	**84.17**	88.75	**97.50**
D+I+Ctx	**64.50**	**73.44**	**75.00**	77.92	**89.17**	96.88

Table 1: Classification Accuracies of CNN-LSTM across various categories (columns) and features (rows). (D = Depth feature, I = Intensity feature, Ctx = Context included)

For each feature type (depth or intensity), we evaluated the gesture recognition accuracy of CNN-LSTM based on the six categories of gestures from CAT 5 to CAT 2. The accuracies are listed for each category in the first two rows of Table 1. The accuracies range from 49.8%(CAT 5)-96.9%(CAT 2). Most of the inter-class confusions occur between (a) taps, blooms, and random gestures and (b) pokes and attentions. We then included context information at each time step by stacking neighboring frames along the channel. For depth features, we used a context window of size 5 (*i.e.*, from $t-2, \cdots, t+2$). Thus, at each time step, the input raw image with context was a tensor of dimension $17 \times 17 \times 5$ instead of a $17 \times 17 \times 1$ tensor without context. Similarly, for intensity features, we used a context window of size 7. The third and fourth rows in Table 1 list the accuracies when context was included. On an average, the increase in accuracy due to context was 2.1% for depth and 10.6% for intensity. The increase in accuracy for intensity was mostly due to the blooms getting classified correctly. Finally, the last row in Table 1 represents the accuracies of the dual-input CNN-LSTM of Fig. 3(b) with context included. The accuracies are in the range 64.5%(CAT 5) - 96.9% (CAT 2). The average increase in accuracy was 10.3% when compared with depth without context. The increase for intensity features with context over without context was 13%. Finally, it is useful to note the performance of some contemporary systems which use optical sensors and deep neural nets. We point to the results reported in [4, DeepPrior in Figs. 7, 8] to predict static hand poses. The frame accuracies reported are 85% and 96% for the ICL and NYU test sets respectively. Although the results are based on static hand poses instead of dynamic and on different datasets, they still allude to potential scope for improvement of our proposed ultrasound system.

5. CONCLUSIONS

We presented a system for end-to-end ultrasound based gesture recognition using a single piezoelectric transducer and an 8-element microphone array. First, we insonified the entire image in one shot, allowing us to achieve high frame rates, enough to capture dynamic gestures in real-time. Next, we obtained ultrasonic images using depth and intensity features. Finally, we recognized gestures using CNN-LSTM networks trained on these ultrasonic images. We reported accuracies in the range 64.5-96.9%, which point to the possible use of the proposed ultrasound system as a low-energy hand-gesture IO interface in mobile and interactive devices.

6. REFERENCES

[1] H. Bai, G. Lee, and M. Billinghurst, "Using 3D hand gestures and touch input for wearable ar interaction," in *CHI Extended Abstracts on Human Factors in Computing Systems*, 2014, pp. 1321–1326.

[2] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, and B. MacIntyre, "Recent advances in augmented reality," *IEEE Computer Graphics and Applications*, vol. 21, no. 6, pp. 34–47, 2001.

[3] S. Mitra and T. Acharya, "Gesture recognition: A survey," *IEEE Trans. Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 3, pp. 311–324, 2007.

[4] J. Supančič, G. Rogez, Y. Yang, J. Shotton, and D. Ramanan, "Depth-based hand pose estimation: data, methods, and challenges," in *Proc. IEEE Int. Conf. Comp. Vision*, 2015.

[5] J. Stühmer, S. Nowozin, A. Fitzgibbon, R. Szeliski, T. Perry, S. Acharya, D. Cremers, and J. Shotton, "Model-Based tracking at 300 Hz using raw time-of-flight observations," in *Proc. IEEE Int. Conf. Comp. Vision*, 2015.

[6] K. Kalgaonkar and B. Raj, "Acoustic doppler SONAR for gait recognition," in *ICASSP*, 2007.

[7] K. Kalgaonkar and B. Raj, "One-handed gesture recognition using ultrasonic doppler SONAR," in *ICASSP*, 2009.

[8] B. Zhu, T. Hazen, and J. Glass, "Multimodal speech recognition with ultrasonic sensors," in *Interspeech*, 2007.

[9] I. Dokmanić and I. Tashev, "Hardware and algorithms for ultrasonic depth imaging," in *ICASSP*, 2014, pp. 6702 – 6706.

[10] J. Capon, "High-Resolution Frequency-Wavenumber spectrum analysis," *Proc. IEEE*, vol. 57, no. 8, pp. 1408–1418, 1969.

[11] I. Tashev, *Sound Capture and Processing, Practical Approaches*, Wiley, UK, 1 edition, 2009, ISBN 978-0-470-31983-3.

[12] M. Thomas, H. Gamper, and I. Tashev, "BFGUI: An interactive tool for the synthesis and analysis of microphone array beamformers," in *ICASSP*, 2016.

[13] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *CVPR*, 2015.

[14] O. Hamid, A.-R. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 10, pp. 1533–1545, Oct 2014.

[15] O. Vinyals, S. V. Ravuri, and D. Povey, "Revisiting recurrent neural networks for robust ASR," in *ICASSP*, 2012.

[16] A. L. Mass, Q. V. Le, T. M. O'Neil, O. Vinyals, P. Nguyen, and A. Y. Ng, "Recurrent neural networks for noise reduction in robust ASR," in *Interspeech*, 2012.

[17] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Joint optimization of masks and deep recurrent neural networks for monaural source separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 12, pp. 2136–2147, 2015.

[18] T. Mikolov, M. Karafiat, L. Burget, J. Cernocky, and S. Khudanpur, "Recurrent neural network based language model," in *Interspeech*, 2010.

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

[20] A. Graves, A. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *ICASSP*, 2013.

[21] D. Yu *et al.*, "An introduction to computational networks and the computational network toolkit," Tech. rep., Microsoft, Redmond, WA, 2014.