SPEAKER LOCALIZATION FOR FAR-FIELD AND NEAR-FIELD WIDEBAND SOURCES USING NEURAL NETWORKS

Güner Arslan[1], F. Ayhan Sakarya[2,3], and Brian L. Evans[1]

[1] Dept. of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712-1084 USA

[2] Dept. of Electronics and Telecommunication Engineering, Yıldız Technical University, 80750 Istanbul, Turkey

[3] Wireless Technology Laboratory, Lucent Technologies, Holmdel, NJ 07733-3030 USA

ABSTRACT

Many applications such as hands-free videoconferencing, speech processing in large rooms, and acoustic echo cancellation, use microphone arrays to track speaker locations in real-time. A speaker is a wideband source which may be in the near field or far field of the array. Current source localization approaches based on neural networks can meet real-time constraints but assume far-field narrowband sources. In this paper, we (1) apply neural networks for determining direction-of-arrival for near-field and far-field wideband speaker localization, and (2) compute the instantaneous cross-power spectra between adjacent pairs of sensors to form the feature vector. We optimized the overall speaker localization system off-line to yield an absolute error of less than 6 degrees at an SNR of 10 dB and a sampling rate of 8000 Hz at each sensor. When performing speaker localization in real-time, the system would require 1 MFLOP/s.

1. INTRODUCTION

Location of a speaker is important information in many microphone array applications. This knowledge, for example, is required to steer a videoconferencing camera, hands-free, to the current speaker. Acoustic echoes and reverberation, which plague speech applications in closed environments, can be eliminated by using microphone arrays and beamforming techniques [1]. In these techniques, the location of the speaker has to be estimated automatically so that the beamformer look angle can be steered to that location [2].

Many source localization algorithms are computationally intensive and difficult to implement in real time [2]. Neural network based techniques have been proposed to overcome the computational problem by exploiting their massive parallelism [3, 4]. However, most of these techniques assume narrowband far-field source signal, i.e. the incoming wave is planar over the array [3, 4, 5]. These assumptions are not always applicable [2]. For example, in videoconferencing, microphones are generally very close to the speaker.

In this paper, we design a system that estimates the direction-of-arrival (DOA) for far-field and near-field wideband sources. The system uses feature extraction followed by a neural network. Feature extraction is the process of removing redundancy from data which will fed in the neural network but keeping the required information. The neural network, which performs the pattern recognition, computes the DOA to locate the speaker. The key insight is the use of the instantaneous cross-power spectrum at each pair of sensors. By the instantaneous cross-power spectrum we mean the cross-power spectrum calculated without any averaging over realizations. This step calculates the discrete Fourier transform (DFT) of the signals at all sensors, finds the frequencies with large magnitudes, and multiplies the DFT coefficients at these frequencies using the complex conjugate of the coefficients in the neighboring sensors. Compared to traditional cross-power spectrum estimation technique which multiplies each pair of DFT coefficients and averages the results, we save a significant amount of computation. Two sensors are enough to calculate the cross-power spectrum and estimate the DOA. Additional sensors increase the effective signal-to-noise ratio (SNR) but do not provide any additional information for DOA estimation. In our simulations, we found that four sensors was the point of diminishing returns.

Section 2 describes techniques for speaker localization. Section 3 explains feature selection and computation. Section 4 discusses the training and testing of our speaker localization technique. Section 5 analyzes the computational complexity of our technique. Section 6 concludes the paper.

2. SPEAKER LOCALIZATION

In locating a speaker, we estimate the DOA of the source source using data received by a uniform linear array of microphones [4]. A far-field assumption is valid if the distance between the speaker and reference microphone is larger than $\frac{2D^2}{\lambda_{\min}}$ [2], where $\lambda_{\min}$ is the minimum wavelength in the source signal and D is the array aperture. If this condition holds, then incoming waves are approximately planar. So, the time delay of the received signal between the first (reference) microphone and the m^{th} microphone is

$$\tau_m = (m-1)\frac{d\,\sin\theta}{c} = (m-1)\tau \qquad (1)$$

where d is the distance between two microphones, θ is the DOA, and c is the velocity of sound in air. So, τ is the delay between any two neighboring microphones, as shown in Fig. 1. In Fig. 1, the dashed lines represent the incoming plane wave, which arrive at the second microphone τ

B. L. Evans was supported by a US National Science Foundation CAREER Award under grant MIP-9702707. G. Arslan was supported by Turkish Government Higher Education Council (YOK) Fellowship administered by Yildiz Technical University.

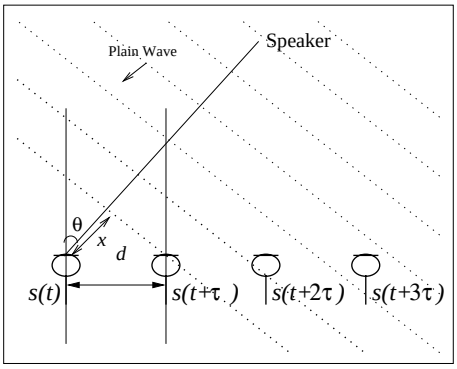

Figure 1: Far-field source location estimation.

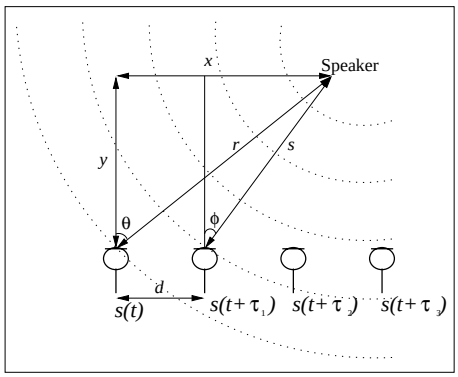

Figure 2: Near-field source location estimation.

seconds before it arrives at the first microphone. The time delay between two microphones is the time required by the plane wave to travel the distance of x, which can be written as $d \sin\theta$. Thus, dividing this distance to the speed of the wave gives the delay as shown in (1). When the speaker is close to the microphone array, the time delay depends not only on d, θ, and c, as in (1), but also on the distance r between the speaker and the microphone array. Therefore, the time delays between microphones are not equal as in the previous case. The time delay from the first microphone to the m^{th} microphone can be written as

$$\tau_m = \frac{r - \sqrt{r^2 - 2\,(m-1)\,r\,d\sin(\theta) + ((m-1)d)^2}}{c} \quad (2)$$

which can be obtained from Fig. 2.

The distance r between the speaker and the first (reference) microphone can be written in terms of x and y as

$$r^2 = x^2 + y^2 \quad (3)$$

The distance from the speaker to the mth microphone is

$$\begin{aligned} s^2 &= (x-(m-1)d)^2 + y^2 \\ &= (x^2+y^2) - 2\,(m-1)\,x\,d + (m-1)^2 d^2 \\ &= r^2 - 2\,(m-1)\,x\,d + (m-1)^2 d^2 \end{aligned} \quad (4)$$

where x can be written in terms of θ and r as $x = r\,\sin\theta$,

$$s^2 = r^2 - 2(m-1)\,d\,r\,\sin\theta + (m-1)^2 d^2 \quad (5)$$

The distance that the wave has to travel between the reference and the m^{th} microphone is

$$r - s = r - \sqrt{r^2 - 2(m-1)\,d\,r\,\sin\theta + (m-1)^2 d^2} \quad (6)$$

The time delay in (2) is obtained by dividing the distance to the velocity of the speech signal in air.

3. FEATURE SELECTION

The multilayer perceptron (MLP) [6] is a feedforward neural network that consists of one input layer, one or more hidden layers, and one output layer. An MLP is capable of approximating any multidimensional mapping with an arbitrarily small approximation error, provided that enough

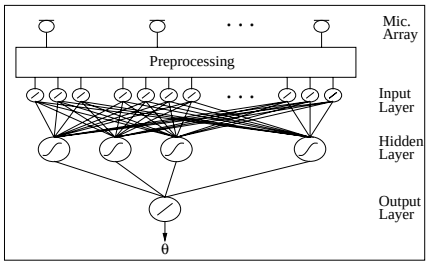

Figure 3: Multilayer perceptron neural network for speaker location. The preprocessing step computes the feature vector to be input the neural network.

hidden neurons are used [6]. Our goal is to compute feature vectors from the array data and use the MLP approximation property to map the feature vectors to the corresponding DOA, as shown in Fig. 3. Ideally, feature vectors

1. could be mapped to the desired output (DOA),
2. are independent of phase, frequency, bandwidth, and amplitude of the source, and
3. could be calculated in a computationally efficient way.

The second item may be relaxed— the MLP only has to be able to eliminate dependencies on the source parameters. Although we compute the feature vectors in discrete time, we justify their use using continuous-time derivations. Assume that $s_m(t)$ is the signal received at the mth microphone and $m = 1$ is the reference microphone ($\tau_1 = 0$). We can write the signal at the mth microphone in terms of the signal at the first microphone as follows

$$s_m(t) = s_1(t+\tau_m) \stackrel{\mathcal{F}}{\rightarrow} S_m(\Omega) = S_1(\Omega)e^{j\Omega\tau_m} \quad (7)$$

The instantaneous cross-power spectrum between sensor m and sensor $m+1$ is defined as

$$\begin{aligned} \phi_{m,m+1}(\Omega) &= S_m(\Omega)S^*_{m+1}(\Omega) \\ &= S_1(\Omega)e^{j\Omega\tau_m}S^*_1(\Omega)e^{-j\Omega\tau_{m+1}} \\ &= |S_1(\Omega)|^2 e^{j\Omega(\tau_m-\tau_{m+1})} \end{aligned} \quad (8)$$

The normalized version of $\phi_{m,m+1}(\Omega)$ is

$$\hat{\phi}_{m,m+1}(\Omega_i) = e^{-j\Omega_i(\tau_m-\tau_{m+1})} \quad (9)$$

1. Calculate the N-point FFT of the received signal at each sensor.
2. For $m = 1, 2, \ldots, M-1$
   (a) Find the K largest FFT coefficients in absolute value for sensor m
   (b) Multiply the K largest FFT coefficients for sensor m with the conjugate of the FFT coefficients at the same indices for sensor $m+1$ to calculate the instantaneous estimate of the cross-power spectrum.
   (c) Normalize all estimates by dividing by their absolute value.
3. Construct a feature vector that contains the real and imaginary parts of the normalized cross-power spectrum coefficients and their corresponding FFT indices.

Figure 4: Preprocessing algorithm for computing a real-valued feature vector of length $(2(M-1)+1)K$, for K dominant frequencies and M sensors.

We would evaluate (9) at frequencies Ω_i for $i = 1, 2, \ldots, K$. This suggests that there exists a mapping from $\hat{\phi}_{m,m+1}(\Omega_i)$ and Ω_i to τ_m for $m = 1, 2, ..., M$, and thus to the DOA θ. Our aim is to use an MLP to approximate this mapping. Fig. 4 gives the algorithm to compute the feature vector.

4. TRAINING AND TESTING

We train an MLP using the fast backpropagation training algorithm [7]. We model the speech signal as a sum of cosines with random frequencies. The array signal at sensor m and time sample n is given by

$$s_m[n] = \sum_{k=1}^{N_s} a_k \cos\left(2\pi\frac{f_k}{f_s}n - \phi_k - 2\pi f_k \tau_m\right) + v[n] \quad (10)$$

where N_s is the number of cosines; f_k is the frequency of the kth cosine; f_s is the sampling frequency at the sensors; ϕ_k is the initial phase of the kth cosine; τ_m is the time delay between the reference microphone ($m = 1$) and the mth microphone; and $v[n]$ is a white Gaussian noise process. Since we are processing speech signals, we assume $N_s = 10$ to model the received speech signal with 10 dominant frequencies, and $f_s = 8000$ Hz.

During training and testing, f_k is uniformly distributed on [200 Hz, 2000 Hz] and ϕ_k is uniformly distributed on $[0, 2\pi]$. We generate training data for θ from -90 to 90 degrees with 5 degree increments resulting in 36 samples of θ. We expect that the MLP will interpolate for frequencies between the increments using its generalization property [6]. Due to the ambiguity of -90 and 90 degrees, we only include -90 degrees. For every θ, we generate 100 independent sets of 128 snapshots each, and then calculate feature vectors. (A snapshot is the vector formed by the data value at the sensors at a particular index of time.) A total of $36 \times 100 = 3600$ input-output pairs are used to train the MLP. Since the backpropagation algorithm can only guarantee a local minimum solution, we repeat training 10 times and choose the best result.

As a performance measure, we calculate the average of the absolute error in degrees over 100 independent tests performed for every θ in one degree steps in the interval from -90 to 90 degrees. We then average these errors to get a single measure for the performance of a particular network in total. In Figs. 5 and 6, most of the error is near the extremes on both sides. This can be explained by rewriting (9) for $m = 1$:

$$\hat{\phi}_{1,2}(\Omega_i) = e^{-j\Omega_i(\tau_1-\tau_2)} = \cos(\Omega_i\tau_2) - j\sin(\Omega_i\tau_2) \quad (11)$$

Since

$$\Omega_i\tau_2 = 2\pi f_i \frac{d\,\sin(\theta)}{c} = 2\pi\frac{d\,\sin\theta}{\lambda_i} \quad (12)$$

and using $d < \frac{1}{2}\lambda_i$ we obtain $\Omega_i\tau_2 < \pi\,\sin\theta$. Thus, when θ is changing in the interval -90 to 90, $\Omega_i\tau_2$ is changing in the interval $-\pi$ to π with the assumption $d = \lambda_i/2$. In (11), $\hat{\phi}_{1,2}(\Omega_i)$ has the same value of -1 for $\Omega_i\tau_2 = \pi$ and $\Omega_i\tau_2 = -\pi$. This reflects the ambiguity at -90 and 90 degrees mentioned earlier. Since the value of $\hat{\phi}_{1,2}(\Omega_i)$ near $\Omega_i\tau_2 = -90$ is close the the value near $\Omega_i\tau_2 = +90$ the MLP is not able to distinguish between these extremes which causes these large errors. Note that the difference between the values of $\hat{\phi}_{1,2}(\Omega_i)$ at the extremes can be increased (the ambiguity decreased) by decreasing d or the range of θ.

Our simulation results confirm that choosing the inter-sensor spacing $d < \lambda_i/2$ decreases the error at the extreme angles and thus the average error (Fig 11). We can say that error in the more useful range of -75 to 75 degrees is actually even smaller than the average absolute error reported in our results. Our simulations show that about 20% of the overall error is outside the range of -75 to 75 degrees.

We train and test the MLP using different numbers of sensors, hidden nodes, snapshots, and dominant frequencies, and different inter-sensor spacing. Figs. 7–11 plot the average absolute error in the DOA estimate vs. one of the parameters. For each figure, we first fix all of the parameters except one and find the optimum value for this parameter. Then, we fix this parameter to the optimum and test the MLP by changing another parameter. Once the second parameter is fixed, we go back and test the network again for the first parameters to make sure the change in the second parameter does not change the optimum value for the first one.

Fig. 7 shows that four sensors is a good choice— using more than four gives diminishing returns. We take $M = 4$. Fig. 8 gives the change in the average absolute error with respect to the number of hidden nodes. The error is high when the network does not have enough neurons for an accurate approximation and when it overmodels the function by using too many hidden neurons. The best choices for the number of hidden nodes are 8 for far-field sources and 10 for near-field sources. We take $L = 10$ as the best choice.

Fig. 9 plots the error vs. the number of snapshots N. The number of snapshots, which is the FFT length in Fig. 4, is plotted for powers of two. The error is nearly constant after 128 snapshots. We take $N = 128$. Fig. 10 shows the error vs. number of dominant frequencies K. The error is

decreasing when K is increased because the network is supplied with more estimates of the instantaneous cross-power spectrum. Since we simulated speech as a signal with 10 dominant frequencies with random amplitudes, K is limited to a value of 10. However, the fairly constant behavior of the error when K is larger than six suggests that on average, only six frequency bins have a large enough magnitude to carry useful information. This is because some of the random amplitudes of the cosines are randomly chosen to be too small or the DFT estimate of these frequencies is too small to count them as dominant.

In many array processing applications, a common rule of thumb is that the optimum inter-sensor spacing d is slightly less than half the minimum wavelength $\lambda_{\min}$ in the signal. Our simulations, we use

$$\frac{\lambda_{\min}}{2} = \frac{c}{2 \cdot f_{\max}} = \frac{340}{2 \cdot 2000} = 0.085\text{m}$$

yet the best choice for the inter-sensor spacing is 0.05 m, i.e. $d = 0.0294\,\lambda_{\min}$. The reason for this behavior was mentioned earlier in this section when discussing the reason of the large errors in the extreme angles (-90 and 90 degrees).

5. COMPUTATIONAL REQUIREMENTS

For N snapshots, K dominant frequencies, M sensors, and L hidden units in the MLP, the number of real multiplications (additions) for speaker localization is the sum of

N-point FFT for M sensors	$2MN\log_2 N$
Instant. cross-power spectral est.	$4KM$
Forward propagation for MLP	$((2M-1)K+1)L$

In addition, $K(M-1)$ real divisions are needed to normalize the instantaneous cross-power spectrum. We implement the nonlinear sigmoid activation function of the MLP as a lookup table. Since f_s/N snapshots are processed each second, the speaker localization system requires 1 MFLOP/s for $M = 4$, $L = 10$, $N = 128$, $K = 6$, and $f_s = 8000$ Hz.

6. CONCLUSION

We develop a speaker localization system for determining DOA angles for near-field and far-field wideband source signals. The system consists of a uniform linear array of sensors, feature extraction, and a neural network. The extracted features are the samples of the instantaneous power spectrum estimated at a set of dominant frequencies corresponding to the maximum FFT coefficients in magnitude. The neural network maps a feature vector to a DOA angle.

We model the speech signal received at the sensors as a sum of cosines with random amplitude, frequency, and phase. We train the neural network for different numbers of sensors, hidden nodes, snapshots, and dominant sinusoidal frequencies, as well as different inter-sensor spacing. We test the neural network for different SNR values. Our results indicate that DOA angle can be estimated with an average error of 2–3 degrees for far-field sources and 3–4 degrees for near-field sources. The largest absolute errors occur near the extremes of -90 and 90 degrees. If the DOA range is limited to -75 to 75 degrees, then the error decreases by 20%.

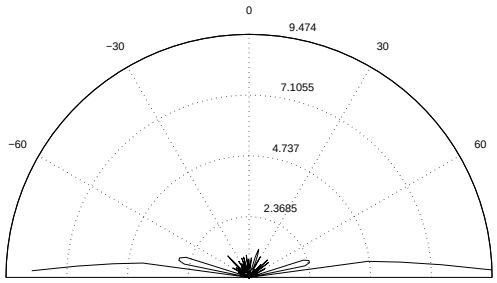

Figure 5: The distribution of the average absolute estimation error in degrees for far-field sources with 4 sensors, 10 hidden units, 128 snapshots, 6 dominant frequencies, and inter-sensor spacing of 0.05 m.

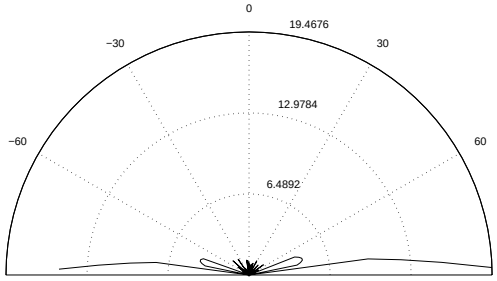

Figure 6: The distribution of the average absolute estimation error in degrees for near-field sources using 4 sensors, 10 hidden units, 128 snapshots, 6 dominant frequencies, and inter-sensor spacing of 0.05 m.

7. REFERENCES

[1] G. Arslan, "Sterefonik Akustik Yankilarin Giderilmesi (Stereophonic Acoustic Echo Cancellation)," Master's thesis, Dept. of Electronics and Telecomm. Eng., Yildiz Technical University, Yildiz, Besiktas 80750, Istanbul, Turkey, July 1996.

[2] M. Jian, A. C. Kot, and M. H. Er, "DOA estimation of speech sources with microphone arrays," in Proc. IEEE Int. Sym. Circ. Sys., vol. 5, pp. 293–6, June 1998.

[3] A. H. El-Zooghby, C. G. Christodoulou, and M. Georgiopoulus, "Performance of radial-basis function networks for direction of arrival estimation with antenna array," IEEE Trans. Antennas and Prop., vol. 45, pp. 1611–17, Nov. 1997.

[4] G. Arslan, F. Gurgen, and F. A. Sakarya, "Application of neural network to bearing estimation," in IEEE Int. Conf. on Elect., Comm., Signal Proc., vol. 2, pp. 647–50, Oct. 1996.

[5] B. Colnet and J.-C. D. Martino, "Bearing estimation with time-delay neural networks," in Proc. IEEE Int. Conf. Acoust., Speech, Signal Proc., vol. 5, pp. 3583–6, May 1995.

[6] S. Haykin, Neural Networks. Prentice Hall, 2nd ed., 1999.

[7] The MathWorks Inc., MATLAB Neural Network Toolbox User's Guide, 1996.

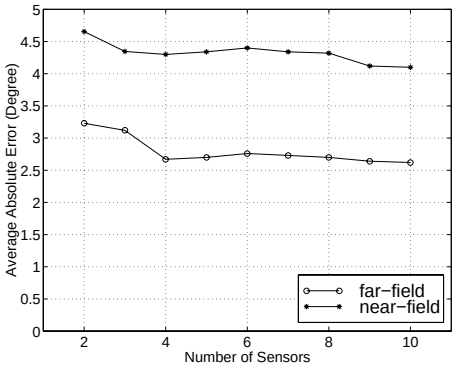

Figure 7: Average absolute error in the DOA estimate vs. the number of sensors using 10 hidden units, 128 snapshots, 6 dominant frequencies, and 0.05 m inter-sensor spacing.

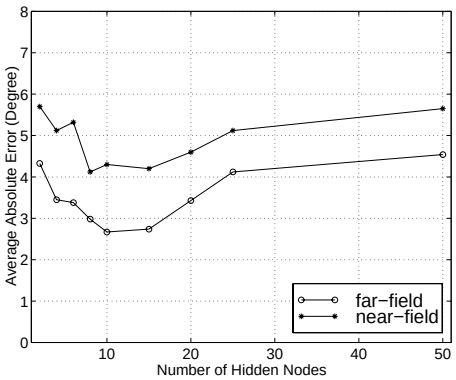

Figure 8: Average absolute error in the DOA estimate vs. the number of hidden nodes using 4 sensors, 128 snapshots, 6 dominant frequencies, and 0.05 m inter-sensor spacing.

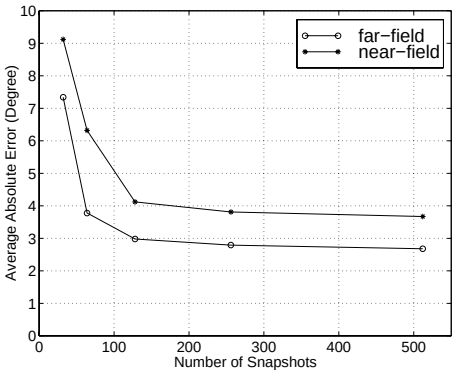

Figure 9: Average absolute error in DOA estimate vs. the number of snapshots using 4 sensors, 10 hidden units, 6 dominant frequencies, and 0.05 m inter-sensor spacing.

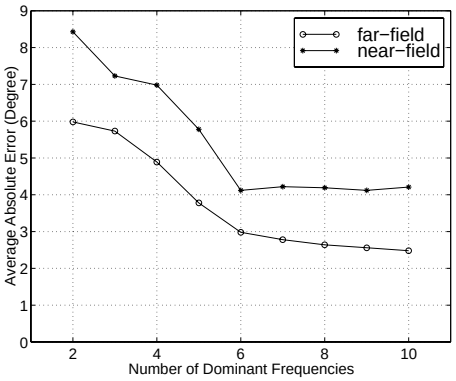

Figure 10: Average absolute error in DOA estimate vs. the number of dominant frequencies using 4 sensors, 10 hidden units, 128 snapshots, and 0.05 m inter-sensor spacing.

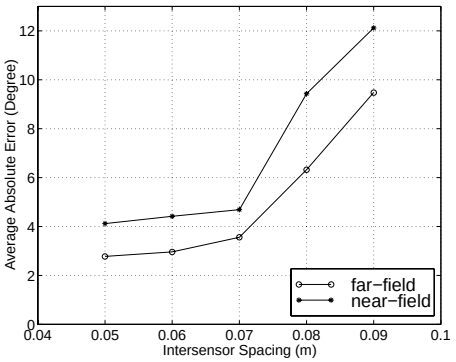

Figure 11: Average absolute error in DOA estimate vs. inter-sensor spacing using 4 sensors, 10 hidden units, 128 snapshots, and 6 dominant frequences.

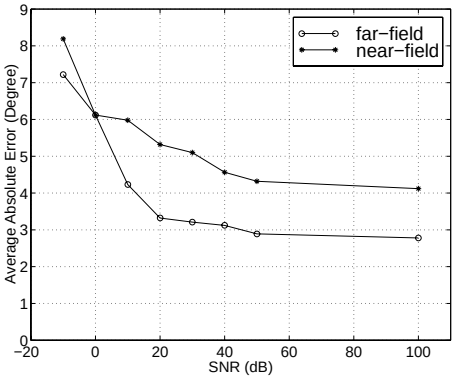

Figure 12: Average absolute error in DOA estimate vs. SNR using 4 sensors, 10 hidden units, 128 snapshots, 6 dominant frequences, and 0.05 m inter-sensor spacing.