

VASLA: Visually Assisted Sound-Localization and Amplification

Xincheng Huang

University of British Columbia
Vancouver, British Columbia, Canada

Chenwei Zhang

University of British Columbia
Vancouver, British Columbia, Canada

Michael Yin

University of British Columbia
Vancouver, British Columbia, Canada

ABSTRACT

In noisy environments, it can be difficult for machines to separate sounds of interest from background sounds. This problem, commonly referred to as the “cocktail party problem” has been a significant area of research over several decades. In this study, we develop VASLA (Visually Assisted Sound-Localization and Amplification), a tool to help alleviate this problem. With VASLA, a user can select an object by clicking on a digital interface that reflects the environment around them. Our system will then amplify the sound source coming from this selected location. The theory of VASLA lies in amplification arising from delay-and-sum beamforming - by calculating the difference in time in which a sound source from a specific location hits different microphones in an array, we can delay the signals from certain microphones to create an amplified sound from that location. Overall, our tool is relatively lightweight, using a camera for visual input and a microphone array for audio input. We evaluate our system with 4 approaches - spectrogram comparison, signal-to-noise (SNR) ratio, word-error rate (WER) analysis, and human perception, finding that it performs better than ground truth audio input.

CCS CONCEPTS

• Human-centered computing → Sound-based input / output.

KEYWORDS

visual localization, sound source augmentation, accessibility

ACM Reference Format:

Xincheng Huang, Chenwei Zhang, and Michael Yin. 2018. VASLA: Visually Assisted Sound-Localization and Amplification. In *Proceedings of ACM Conference (Conference’17)*. ACM, New York, NY, USA, 9 pages. <https://doi.org/10.1145/1122445.1122456>

1 INTRODUCTION

Humans are constantly faced with sounds in their environments. Many of these sounds compete against each other, continually vying for the aural attention of any person. Within such overwhelmingly noisy environments, how can a person concentrate on the sound coming from a specific object of interest? Humans have the capability to do this through a phenomenon commonly referred to as the “cocktail party effect” - the psychoacoustic ability for humans to follow a conversation amidst noisy environments [3, 15, 19, 28, 31].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Conference’17, July 2017, Washington, DC, USA

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-XXXX-X/18/06...\$15.00

<https://doi.org/10.1145/1122445.1122456>

However, despite the relative ease in which humans are able to do this, it has been a challenge for researchers to pinpoint what particular biological aspect affords this ability. Research has showed that this phenomenon is related to a more general concept called streaming, in which humans have the intrinsic ability to group together sounds from one source and separate them from other sources [28]. Although the biomechanics of this human ability is not fully understood, research has showed that humans can separate sounds based on cues such as location [2, 7], frequency [5], onset and offset times [5], and amplitude modulation [4].

The uncertainty in understanding the biomechanics behind the cocktail party effect has consequently led to challenges for researchers to artificially replicate such behaviour in machines - the “cocktail party problem” [8, 16, 24]. A innumerable amount of research has been done in this area, aiming for the separation of sources in noisy environments. In particular, researchers have attempted to separate voices in environments of multiple speakers [10, 26, 44, 45] or in environments of high noise [6, 23, 40], through a variety of signal processing and machine learning techniques. Other researchers have looked into the problem of separating differently-pitched instruments from a musical composition [22, 29]. On occasion, researchers will augment a purely auditory approach for separation with visual information in the form of audio-video inputs [47, 48]. At a more general, context-independent level, researchers have looked into separating any arbitrary sounds within an environment, a problem commonly posed as “universal sounds separation” [18, 37].

In this study, we present a novel, context-independent approach at addressing the “cocktail party problem” - amplifying the sound coming from a specific location of interest. Instead of separating out the noisy signals, we instead boost the sound coming from a specific source, helping users hear the location of interest more clearly. Our theory revolves around adaptive beamforming and the constructive and destructive interference of sound signal frequencies in different phases. In our implementation, users are able to select a sound source of interest from a 2D display. After localizing the source within the 3D environment, we use adaptive beamforming to amplify the sound coming from that location. We test our application within multiple configurations of noisy environments, and evaluate the validity of our approach using four core metrics - human subjective perception, spectrogram and waveform comparison, signal-to-noise (SNR) ratio, and speech-to-text error rate. We then discuss the potential of our system in practice applications, such as its affordances for accessibility and audio mixing.

2 RELATED WORK

2.1 Separating Sounds in Noisy Environments: The Cocktail Problem

As established, the cocktail party problem refers to the difficult challenge of understanding how humans can pick out sounds in noisy environments [16, 24]. In recent years, the cocktail party problem has since become a problem of sound source separation, a topic of research that has been explored by several researchers in various ways [12]. Historical research has typically primarily focused on techniques such as matrix factorization. For example, Virtanen used non-negative matrix factorization to minimize the reconstruction error after separation into individual sound matrices [38]. Zhong et. al. use the singular value analysis to ensure that the sounds are separated with minimal distortion [49]. The rise of machine learning has also opened the door for such approaches in sound source separation problems. A variety of different models have been used in an attempt to address sound source problems, for example, Demir et. al. used a CNN model to classify environmental sound sources [9] and Luo et. al. used a modified RNN model to separate out speech signals [22]. Many other models take advantage of new techniques in machine learning, such as deep learning, to better address the problem [35, 39]. However, many machine learning tasks are limited to the data that is used as input for learning, commonly speech or music. In contrast, the attempt to separate any arbitrary number of sound sources in an environment is referred to as universal sound separation [18]. Although researchers such as Tzinis et. al. and Li et. al. have attempted to address this challenging area [21, 37], there exist perpetual challenges such as a lack of data [41]. Overall, we present our work as a signal processing approach to separating between a sound of interest and background noise, drawing from knowledge from prior work on sound source separation. Our approach is context-independent and learning-independent, requiring no past training data or knowledge of environment.

2.2 Sound Source Localization

Within our application, we take advantage of location-based information as a method of amplifying the sound source coming from that location. Thus, we look at sound source localization within our literature review, aiming to better understand how machines can obtain this location-based information from audio signals. Much work has been done in this area, with many researchers having applied a variety of different approaches in addressing this problem [20, 32, 34]. For example, time differences of arrival (TDOA) approaches take advantage of the fact that sound travels, thus the time at which a sound reaches microphones spatially separated in space is different [25, 30, 42]. Pavlidi et. al. show the potential for TDOA approaches for high accuracy and low computation complexity [30], and Miao et. al. use show that advanced additions to TDOA methods can track moving sound sources, one of the classical limitations of such approaches [25]. An alternative approach to TDOA is using frameworks such as beamforming and maximum likelihood calculations involving eigenvalue decomposition of sound and source matrices [27, 46]. Research into this approach has shown the benefit

in which such algorithms can apply to unknown directional patterns of microphones. Lastly, with the rise in popularity of machine learning technologies in recent years, researchers have applied such technologies to the application of sound source localization. Yalta et. al. show that end-to-end deep residual methods can surpass the accuracy of classical signal classification methods, and can also be robust to challenging conditions through exploiting time delay [43]. Sun et. al. create a probabilistic neural network that allows for the localization of sound within adverse environments with high reverberation and background noise [36]. Ultimately, we use the information from past research in sound source localization as a motivator within our own application, to understand how localization can work together with separation to amplify the sound source from a specific location.

3 SYSTEM OVERVIEW AND IMPLEMENTATION

3.1 Hardware Setup and System Architecture

Our system is consist of two parts: target sound localization and beamforming. There could exist multiple competing sound sources in an environment. Here, we refer to the sound a user is interested in as the “target sound”, and refer to all the other competing sounds as “noises”. For target sound localization, we implemented with Processing 4.0¹ a simple interface that can play a recorded video captured by a built-in FaceTime HD Camera on a 16-inch Macbook Pro 2019². Such an interface allows user to identify a target sound source by clicking on a specific location on the camera feed. The localization module will translate the coordinate of the pixel selected to the position in 3D using Monodepth2 [14]. For the beamforming module, we used a SeeedStudio ReSpeaker USB Mic Array³, which is consist of 4 microphones installed on the corners of a 4.95 centimeter square. We assume that the position of the microphone array and the sound sources will remain stationary throughout the recordings. Once the user identifies the target sound source, our application calculates the distance between the each microphones and the target sound source, and performs beamforming. Beamforming will boost the sound coming from the target sound source and suppress all the other noises. At completion of beamforming, our application substitutes the original sound of the video with the beamform-processed sound track once beamforming is completed.

3.2 Depth Calculation and Pixel Localization

Once the user has clicked on a pixel within the 2D display, our application estimates the 3D analogue of the position through calculating the depth of the pixel, and then subsequently calculating the position. In order to calculate the depth of the pixel, we use a pre-existing open-source depth calculation implementation called Monodepth2, developed through the research of Godard et. al. [13, 14]. Monodepth2 is an open-source Python software that contains a PyTorch implementation for the training and testing of depth estimation models. This implementation uses a self-supervised monocular neural network training model that estimates depth based on inputs. In comparison to similar implementations, Monodepth2

¹<https://processing.org>

²https://support.apple.com/kb/SP809?locale=en_US

³<https://wiki.seeedstudio.com/ReSpeaker-USB-Mic-Array>

performs well even under occlusion and motion. Ultimately, we can use the pre-trained Monodepth2 model by passing an input of an image of an environmental scene, and the model returns an output of metric depth at each pixel of the scene.

After calculating the depth of each pixel in the image, it is simple to extract out the depth of the pixel of interest. We represent each axis of the 2D pixel as pixel coordinates (x, y) in the range $[-1, 1]$. Thus, the top-left most pixel is represented as $(-1, -1)$, and the bottom-right most pixels is represented as $(1, 1)$. We combine the pixel information with its depth information and the intrinsic parameters of the camera to calculate the 3D location of the pixel, represented by geometric (x', y', z') coordinates, with the origin being the camera lens. In particular, we use the field-of-view (FOV) of the camera (54° for the Apple FaceTime HD Camera) to calculate the projection of the pixel in 3D. We know that z' is depth, thus, we use this to calculate x' and y' . A visual representation of this calculation is shown in Figure 1.

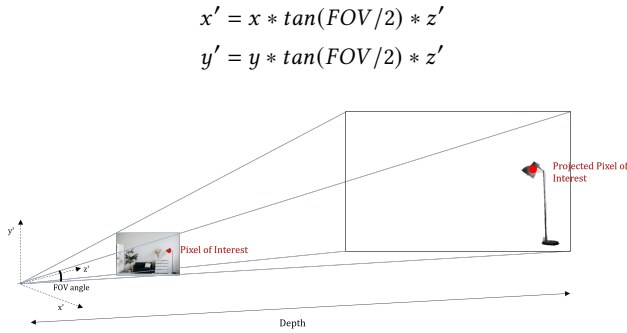


Figure 1: Calculation of Depth from Pixel Data, Depth Information, and Camera Parameters.

3.3 Beamforming

After estimating the 3D position of the sound of interest, our application performs beamforming on the sound channels collected by the mic-array to amplify target sound. Beamforming is an approach that steers a sensor array and maximizes its response toward a target location. Specifically for our application, we implemented delay-and-sum beamforming, which is the most straight-forward version of beamforming that can be performed in almost real-time. Delay-and-sum beamforming is based on the fact any wave-front of sound arrives at microphones at different location at a different time. Therefore, the waveforms obtained by the microphones on a mic-array will be similar but show different delays and phases. Such phase differences can be calculated knowing the positions of the microphones, sound sources, and the speed of sound. Therefore, it is possible to align the phases of the waveforms obtained at different microphones according to the location of the target signal. Adding up the aligned waveforms will cause the target sound signal to construct and the noises coming from elsewhere (i.e. noises and interference) to randomly construct or deconstruct. The resulted signal after beamforming will have the target sound boosted while suppressing any other noises. Figure 2 represents a graphical interpretation of delay-and-sum beamforming.

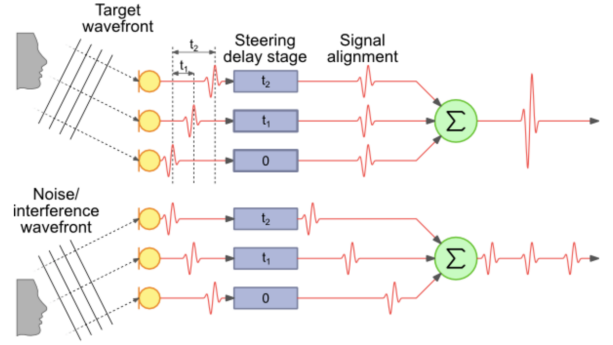


Figure 2: Delay-and-sum beamforming: the top half of the figure shows the effect of delay-and-sum beamforming on the signal of interest. Knowing the distance between of the target signal source and the microphones, the phases are aligned such that the target signal will construct. The bottom half of the figure shows the effect of delay-and-sum beamforming on noises. Since the signals are aligned according to a distance irrelevant to the noises and interference, the noise signals will destruct. Therefore, the resulted signal will boost the target signal and suppress the noise.

3.4 Simulations

Besides implementing the system architecture as described above, we also replicate the major parts of the architecture in a simulated environment with Pyroomacoustics [33]. Pyroomacoustics is a Python package for simulating acoustic effects, which allows us to simulate environments with no reverberation and simulate microphone arrays with various sizes and number of microphones. Our system architecture in simulation shares the same delay-and-sum beamforming techniques as described above but does not implement sound source localization as the simulation has direct access to the locations of the simulated sound sources. A majority of our experiments and evaluations are conducted in simulation due to the limited access to the silent sound studio and more effective mic-arrays. With the simulated environment, we are able to evaluate the beamforming performance with different microphone configurations, different numbers of microphones, and different target sound and noise positions, which we will elaborate in the following sections.

4 EXPERIMENTS AND EVALUATION APPROACHES

4.1 Experiments

We conducted two sets of experiments for evaluation, one in simulation and one in real-life. For simplicity, we constraint our experiments in ideal environments with minimal multi-path effects and reverberations. Also, we only tested our application's performance in terms of amplifying one target sound while suppressing one competing interference.

For the experiment that we conducted in simulation, we simulated a $10 \times 10 \times 10$ room with no reverberation. We first conducted a pilot experiment with different simulated mic-array configurations

with different number of microphones and microphone placements. Throughout the experiments for mic-array configurations we found that the distance between adjacent microphones matters more than the number of microphone used, and the mic-array that leads to the best performance is a mic-array with 6 microphones arranged linearly with 20 centimeters between adjacent microphones. In the subsequent simulated experiments we conducted, we stick to this linear 6-mic configuration.

Using the simulated 6-mic configuration, we first explored several sound-interference combination. The purpose of this experiment was to observe how the nature of sound will affect the performance of beamforming and evaluate the robustness of our application. We summarize the different audio clips we used and the sound-interference combinations we tested respectively in Table 1 and Table 2. For all the sound tracks we used, we pre-processed them in Adobe Audition⁴ such that they have similar powers. Besides different sound-interference combination, we also explored how the locations of sound and noises will affect the performance of beamforming. We placed pairs of sounds and noises at different positions in the simulated room and observed the results returned by beamforming.

Audio Clip	Description
Read1	An excerpt from an audio book <i>A Little Princess</i>
Read2	An excerpt from an audio book <i>A Little Princess</i> (different from read1)
Background Sound	An audio clip of noises in a restaurant
Flute	A sound clip of a flute playing

Table 1: Description of audio clips used in experiments, and each clip lasts for 1 minute.

Test Cases	Target Sound	Noise
Case 1	Read1	Read2
Case 2	Read1	Background Sound
Case 3	Flute	Background Sound

Table 2: We created three test cases with different sounds and noises competing with each other. This table summarizes the three test cases.

For the real-life experiment, using the hardware setup we described in section 3.1, we recorded a video with two sound sources located at two difference positions playing two different sound tracks ("Read1" and "Background Sound") simultaneously. We conducted this experiment in a sound studio with sound-absorbing walls for minimal reverberations. Figure 3 shows one frame of the recorded video.



Figure 3: A Frame from the Interface. The position of the target sound and noise are respectively noted in red.

4.2 Evaluation Approaches

We used four different approaches to evaluate the performance of our system, including human perception, waveform and spectrogram comparison, signal-to-noise ratio (SNR) analysis, and word-error-rate (WER) analysis. For the real-life experiments we conducted in the sound studio, the target sound amplification turned out to be inaudible and we can only tell the effect of beamforming through the SNR we calculated. For the simulated experiments, the performance of sound amplification after beamforming is relatively more significant and thus we applied all four evaluation approaches.

4.2.1 Human Perception. For the human perception analysis we studied whether the sound amplification by beamforming is audible for human ears. A formal study should be conducted on participants recruited outside the project team members. However, for the purpose of a course project, we rely on the authors to decide whether the difference before and after beamforming is significant. The authors listened to three different audio clips including the original sound tracks (i.e. the target sound and the interfering sound), the unprocessed sound mixture, and the sound mixture after beamforming. We then subjectively tell the whether beamforming has led to an audible amplification of the target sound.

4.2.2 Waveform and Spectrogram Comparison. Second evaluation approach is to plot the waveform and spectrogram of the original target sound, unprocessed sound mixture, and the sound mixture after beamforming, and visually compare these plots. The similarity between the plots corresponding to the target sound and the sound mixture after beamforming reflects the performance of our audio amplification application. Generally speaking, a higher similarity implies a stronger amplification effect achieved by beamforming.

4.2.3 Signal-to-noise Ratio Analysis. Besides the above two subjective metrics, we calculated the SNR to quantitatively evaluate the performance of our beamforming application. SNR is a measure defined as the ratio of the power of sound to the power of noise

⁴<https://www.adobe.com/>

as in Eq. 1, where the powers are calculated as the square of the root-mean-squared (RMS) amplitude over the signals. In practice, a way to estimate the power of sound and noise respectively from the sound mixture is needed to calculate SNR. In our experiments, such an estimation is simple with known original unmixed sound and noise. Recall that our beamforming function takes a target sound and a noise as input, and steer itself towards the target sound by aligning the phases according to the distance between the target sound source and the microphones. Since the process of delay-and-sum beamforming only involves linear addition, we can obtain the target sound after beamforming by applying same beamforming but with the noise replaced with a silent sound track. The noise suppressed by beamforming can be similarly obtained by beamforming with the target sound replaced with a silent sound track.

$$\text{SNR} = 10 \cdot \log_{10}\left(\frac{P_{\text{signal}}}{P_{\text{noise}}}\right) \quad (1)$$

4.2.4 Word-error-rate Analysis. Our last evaluation approach applies to having human speeches amplified by beamforming. We applied a speech-to-text model on both the unprocessed sound mixture and the audios after beamforming and compare the WER of the transcribed scripts, which is the standard measure for evaluating speech-to-text transcription. The speech-to-text application we used for this evaluation is a pre-trained Wave2Vec model [1], which is similar to the Bidirectional Encoder Representations from Transformers (BERT). An automatic speech recognition evaluation system [17] is then utilized for the WER calculation. The WER results provide a straightforward representation of the performance of application. In general, the speech-to-text model should perform better on a sound mixture with the target sound (i.e. a speech in this case) boosted sufficiently, and thus leading to a lower WER.

5 RESULTS

We now present the results from our experiments. We evaluated the results from the simulated experiments with all four approaches as mentioned in the previous section. However, due to the limited number of experiments we are able to run in real-life and the limited beamforming power our hardware provides, we will only evaluate the real-life experiments with human perception and SNR.

5.1 Beamforming Performance in Simulation

5.1.1 Perceptual Results. Evaluating with human perception, all of us found that beamforming in simulation is able to boost the target sound over the noise significantly in all test cases. This result shows our beamforming application works as expected in different environments regardless of the kinds of signals and noises. Nevertheless, perceptually, the extent to which the target sound was boosted varied across the test cases. We found that case 3 (“flute” vs. “background sound”) has the most significant target sound amplification while case 1 (“Read1” vs. “Read2”) has the least. It appears that beamforming is better at amplifying sounds with stable and more consistent frequencies such as flute and reading, while suppressing sounds closer to white noises.

5.1.2 Waveform and Spectrogram Comparison. Figures 4, 5, 6 compare waveforms and spectrogram of the original target sound, sound mixture after beamforming and the original sound mixture in test

cases 1, 2, and 3. It is obvious that in figure 2 and figure 3, which corresponds to case 2 (“read1” vs. “background”) and case 3 (“flute” vs. “background”), the waveforms and spectrograms of the sound mixtures after beamforming become similar to that of the original target sound. This shows that beamforming significantly brings the sound mixture closer to the target sound. In figure 1, which corresponds to case 1, however, the same similarity is less obvious. This result again shows that our beamforming application is better at amplifying sounds against background noises similar to white noises. In contrast, in cases where two clear speeches are competing, a more advanced approach might be needed to improve upon the performance of pure beamforming.

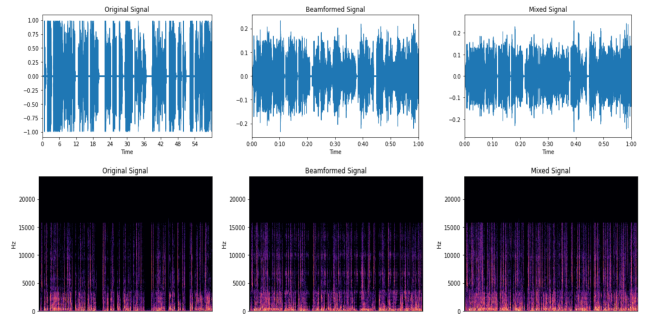


Figure 4: Plots of the sound waveform (top panel) and spectrograms (bottom panel) for experimental case 1 (“Read1” vs. “Read2”).

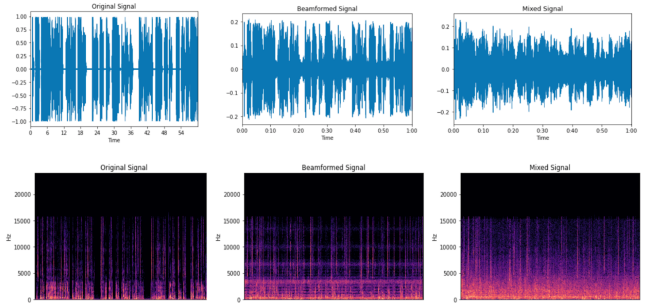


Figure 5: Plots of the sound waveform (top panel) and spectrograms (bottom panel) for experimental case 2 (“Read1” vs. “Background Sound”).

5.1.3 Signal-to-noise Ratio (SNR). To quantify the performance of the proposed model, we first analyzed the signal-to-ratio (SNR), whose results are shown in Table 3 and Figure 7. From the figure, all the SNR scores of sound mixture after beamforming are positive and larger than the original sound mixtures which are all negative. In cases 2 and 3, we can see that beamforming provides about 8 decibels of SNR improvement. Relating back to what we observed in human perception and waveform/spectrogram observation, this is a relatively satisfactory result. In case 1, however, beamforming only provided an SNR improvement of about 2.4 decibels, which

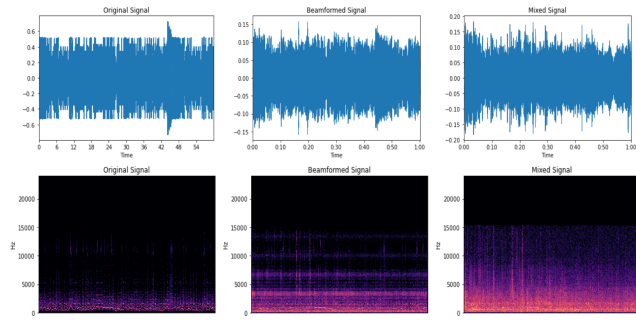


Figure 6: Plots of the sound waveform (top panel) and spectrograms (bottom panel) for experimental case 3 ("Flute" vs. "Background Sound").

again shows that pure beamforming may find it more challenging to boost a target sound against a clear and frequency consistent voice.

	SNR before beamforming (dB)	SNR after beamforming (dB)
Case 1	-0.804	1.586
Case 2	-1.155	7.09
Case 3	-0.493	7.752

Table 3: SNR scores of test case 1, 2, and 3 before and after beamforming. Observe that in all cases the SNR of the mixed sound has a negative SNR, indicating that the noise has a slightly higher power than the target sound. Beamforming significantly amplifies the target sound in case 2 and 3, and mildly in case 1.

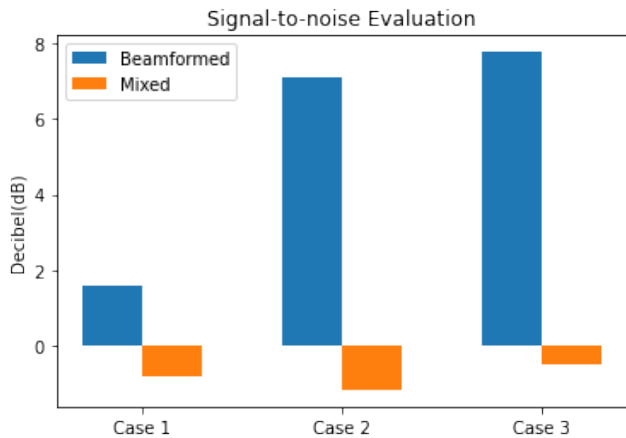


Figure 7: Histogram of signal-to-ratio (SNR) results for experimental cases 1, 2, and 3. Blue and orange bars refer to SNR values of after-beamformed and directly-mixed signals, respectively.

5.1.4 Word-error-rate (WER). Beamforming and sound separation techniques are often seen in improving machine translation in noisy environments. Therefore, it makes sense to evaluate our first and second test cases using WER, which is a common metric to evaluate the performance of speech recognition or machine translation systems. The WER scores in cases 1 and 2 are summarized in Table 4 and plotted in Figure 8, and note that a lower WER means better speech recognition performance. As expected, the WER score for speech recognition on the sound mixture after beamforming are much lower compared with the same score on the original sound mixture, demonstrating that the beamforming application has a significant effect in improving speech recognition in a noisy environment. It is worth noticing that the WER is 100% when mixing a voice audio with a background noise, pointing that in this case the machine is not able to recognize anything. After applying beamforming, the WER score has improved considerably. The machine can recognize the speech context and the final WER is reduced to 30%. Moreover, even if mixing a voice audio with another voice clip, the accuracy would also achieve a high score with ca. 60% as shown in case 1. This shows that even for test case 1, which we have shown that beamforming achieved a relatively lower performance, it is still able to provide considerable performance improvement for speech recognition and machine translation.

	WER(%) before beamforming	WER(%) after beamforming
Case 1	81.4207650273224	38.94736842105263
Case 2	100	30.601092896174865

Table 4: WER scores of test cases 1 and 2. In both cases our beamforming application provides a signification improvement for speech recognition.

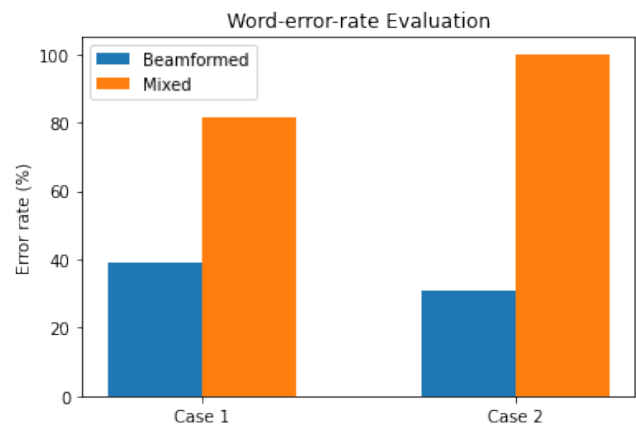


Figure 8: Histogram of word-error-rate (WER) results for experimental cases 1 and 2. Blue and orange bars refer to WER values of after-beamformed and directly-mixed signals, respectively.

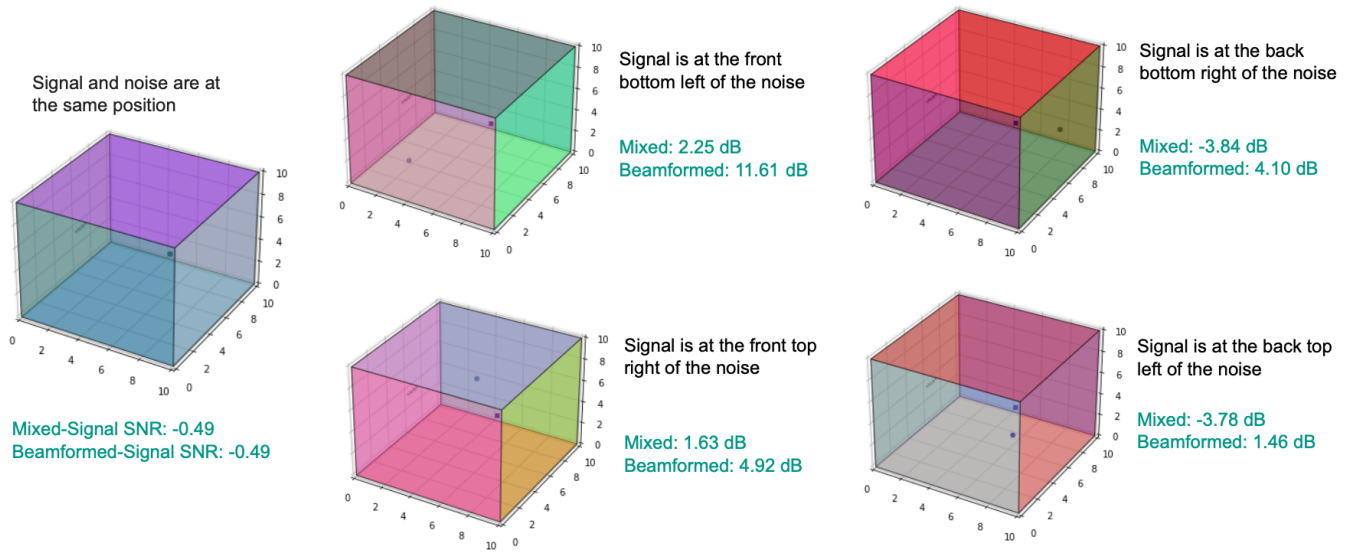


Figure 9: Each cube represents a simulated room with a line of microphones, a fixed noise source, and a voice audio "read1" with various positions. The relative positions and SNR scores are shown in the top right and bottom right of each graph, respectively.

5.2 The Effect of Sound Positions

We also studied how the positions of sounds will affect the performance of beamforming. As shown in Figure 9, the leftmost figure shows the sound of interest and noise at the same position, and the SNR scores are the same, implying the beamforming model does not work in this case. This result makes sense and proves that the designed model is correct as the principle of beamforming is based on the different delays and phases, thus the same position leads to no phase difference and therefore the beamforming does not work. According to the SNR results of the rest of the four figures, it can be concluded that if the signal (sound of interest) is close to the noise source, the SNR different is small indicating the sound amplification is limited, however, if putting the signal far away from the noise, a larger SNR different is obtained and consequently the beamforming effect is enhanced. We can see that the best SNR boosts with beamforming is achieved when the sound and noise are distant in angle. We hypothesize that the angular distance between sound and noise is the key feature that decides the performance of beamforming in terms of amplifying a sound over a noise.

5.3 Beamforming Performance in Real-life

Although our application performs relatively well in simulations, the results of our experiments in real-life is less than satisfactory. We conducted the test case 2 ("read1" vs. "background sound") in the sound studio and find that the sound amplification for the target sound achieved was perceptually inaudible. To see whether the application had any effect at all, we calculated the SNR improvement the beamforming application provided and obtained a result of 0.8 decibels. We attribute such a low performance to three potential reasons. First, our interface based on Monodepth2 [14] does not provide a very precise distance measurement in a small environment as the sound studio. The error of Monodepth2 we observed

in our test environment is about 0.5 - 1.0 meter. Second, we are limited by the microphone array that is available to us. Unlike in simulations, where we are free to simulate any microphone array, in reality the microphone array we use is very small in size, which does not provide a significant phase difference for beamforming to work well. Lastly, the positioning we chose was not optimized for beamforming. As we mentioned, the performance of beamforming is closely related to the angular distance between the sound and noise, yet in our real-life experiment we were not able to place them distant enough from each other.

6 DISCUSSION AND LIMITATIONS

Through our simulated experiments we have found that our application works well in terms of amplifying sounds of interest in a noisy environment. Although the results of our experiments in a real-life sound studio is less than ideal, yet with a microphone array that is larger in size and number of microphones, we believe that in an ideal environment with minimal reverberations, our application should be able to achieve results comparable to what we have observed in simulations.

Our application can have various potential applications, many of which are related to past research in sound source separation. For instance, we can build upon our beamforming application to help users that experience hard of hearing to better hear sounds. By using our application, such users can amplify the sound coming from a specific sound source, which may be far away and may be competing with sources of noise. Extensions to make such an application more possible may be to incorporate the use of a augmented reality (AR) headset, and gaze tracking in order to help those with impaired hearing continually amplify the sounds coming from the sources they are visually focusing on.

Another possible application of our application is in video and audio editing. In particular, editors can use our application as an artistic tool to boost the sound coming from specific video locations post-hoc. For example, within a music video involving an orchestra, one may want to boost the sound coming from a specific violin. This could potentially have promising applications in music learning and practice.

One limitation within our implementation was the use of a pre-trained Monodepth model for our depth calculation. The existing Monodepth2 model that was used in our application was trained on the KITTI dataset [11], which is an environmental dataset often used in autonomous vehicle applications. However, our application is more likely to be used in indoor environments. Thus, it may have been much more accurate to train on a more representative dataset of room environments. However, the limited availability of such a dataset, and the difficulty of creating such a dataset through taking photos and extracting depth information, veered us away from such an approach. However, with more time, this may have been a more valid method to improve the depth estimate and localization of a clicked position of an image.

Another limitation was the small set of test cases and audio clips used in the experiments. In the interest of time and effort, we used a smaller test set. However, in efforts of achieving generalizability for our application, we would need to test the set on more signals and sets. It is also necessary to extend our application to more realistic environments with more severe reverberation and more than two competing sounds.

7 CONCLUSION

In conclusion, we present VASLA, a system to allow for auditory augmentation of a specific sound source in an environment, assisted through localization using a visual source. We take advantage of tools such as delay-and-sum beamforming and image-depth localization in order to generate our complete system, which runs solely using a camera and microphone array. Users can select a location on a digital image of an environment, and the auditory output will amplify the sound source coming from that location. We evaluate our system using both a real-life audio array, as well as a simulated audio array. Our evaluation is performed on 4 metrics of personal perception, waveform and spectrogram comparison, SNR, and WER, and we find that the simulated system generally performs much better than the ground truth. We present discussion on the possible consequences of our results, demonstrate potential applications of the research, and mention the limitations within our methodology. Overall, we believe VASLA can be a useful contribution to the immense research currently done in attempting to address the “cocktail party problem”.

ACKNOWLEDGMENTS

To Robert Xiao and Abi Kuganesan for helping us with the project.

REFERENCES

- [1] Meta AI. 2020. Wav2vec 2.0: Learning the structure of speech from raw audio. <https://ai.facebook.com/blog/wav2vec-20-learning-the-structure-of-speech-from-raw-audio>.
- [2] Guillaume And  ol, Anne Guillaume, Christophe Michey, Sophie Savel, Lionel Pellioux, and Annie Moulin. 2011. Auditory efferents facilitate sound localization in noise in humans. *Journal of Neuroscience* 31, 18 (2011), 6759–6763.
- [3] Barry Arons. 1992. A review of the cocktail party effect. *Journal of the American Voice I/O Society* 12, 7 (1992), 35–50.
- [4] Albert S Bregman, Jack Abramson, Peter Doehring, and Christopher J Darwin. 1985. Spectral integration based on common amplitude modulation. *Perception & Psychophysics* 37, 5 (1985), 483–493.
- [5] Albert S Bregman and Steven Pinker. 1978. Auditory streaming and the building of timbre. *Canadian Journal of Psychology/Revue canadienne de psychologie* 32, 1 (1978), 19.
- [6] Zhuo Chen, Takuya Yoshioka, Liang Lu, Tianyan Zhou, Zhong Meng, Yi Luo, Jian Wu, Xiong Xiao, and Jinyu Li. 2020. Continuous speech separation: Dataset and analysis. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 7284–7288.
- [7] E Colin Cherry. 1953. Some experiments on the recognition of speech, with one and with two ears. *The Journal of the acoustical society of America* 25, 5 (1953), 975–979.
- [8] Seungjin Choi and Andrzej Cichocki. 1997. Adaptive blind separation of speech signals: Cocktail party problem. In *International conference on speech processing*. 617–622.
- [9] Fatih Demir, Muammer Turkoglu, Muzaffer Aslan, and Abdulkadir Sengur. 2020. A new pyramidal concatenated CNN approach for environmental sound classification. *Applied Acoustics* 170 (2020), 107520.
- [10] Jun Du, Yanhui Tu, Yong Xu, Lirong Dai, and Chin-Hui Lee. 2014. Speech separation of a target speaker based on deep neural networks. In *2014 12th International Conference on Signal Processing (ICSP)*. IEEE, 473–477.
- [11] Andreas Geiger, Philip Lenz, and Raquel Urtasun. 2012. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition*. IEEE, 3354–3361.
- [12] Robert Gilkey and Timothy R Anderson. 2014. *Binaural and spatial hearing in real and virtual environments*. Psychology Press.
- [13] Cl  ment Godard, Ois  n Mac Aodha, Michael Firman, and Gabriel J Brostow. 2019. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 3828–3838.
- [14] Cl  ment Godard, Ois  n Mac Aodha, Michael Firman, and Gabriel J Brostow. 2019. Digging into Self-Supervised Monocular Depth Prediction. (October 2019).
- [15] Monica L Hawley, Ruth Y Litovsky, and John F Culling. 2004. The benefit of binaural hearing in a cocktail party: Effect of location and type of interferer. *The Journal of the Acoustical Society of America* 115, 2 (2004), 833–843.
- [16] Simon Haykin and Zhe Chen. 2005. The cocktail party problem. *Neural computation* 17, 9 (2005), 1875–1902.
- [17] Toshish Jawale. 2019. Automatic Speech Recognition (ASR) Evaluation. <https://github.com/symb lai/speech-recognition-evaluation>.
- [18] Ilya Kavalov, Scott Wisdom, Hakan Erdogan, Brian Patton, Kevin Wilson, Jonathan Le Roux, and John R Hershey. 2019. Universal sound separation. In *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 175–179.
- [19] Paul Kuyper. 1972. The cocktail party effect. *Audiology* 11, 5-6 (1972), 277–282.
- [20] Jeroen Lanslots, Filip Deblauwe, and Karl Janssens. 2010. Selecting sound source localization techniques for industrial applications. *Sound and Vibration* 44, 6 (2010), 6.
- [21] Han Li, Kean Chen, and Bernhard U Seeber. 2021. Auditory Filterbanks Benefit Universal Sound Source Separation. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 181–185.
- [22] Yi Luo, Zhuo Chen, John R Hershey, Jonathan Le Roux, and Nima Mesgarani. 2017. Deep clustering and conventional networks for music separation: Stronger together. In *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 61–65.
- [23] Shoji Makino, Te-Won Lee, and Hiroshi Sawada. 2007. *Blind speech separation*. Vol. 615. Springer.
- [24] Josh H McDermott. 2009. The cocktail party problem. *Current Biology* 19, 22 (2009), R1024–R1027.
- [25] Feng Miao, Diange Yang, Rujia Wang, Junjie Wen, Ziteng Wang, and Xiaomin Lian. 2014. A moving sound source localization method based on TDOA. In *43rd International Congress on Noise Control Engineering*.
- [26] Eliya Nachmani, Yossi Adi, and Lior Wolf. 2020. Voice separation with an unknown number of multiple speakers. In *International Conference on Machine Learning*. PMLR, 7164–7175.
- [27] Keisuke Nakamura, Kazuhiro Nakadai, Futoshi Asano, Yuji Hasegawa, and Hiroshi Tsujino. 2009. Intelligent sound source localization for dynamic environments. In *2009 IEEE/RSJ international conference on Intelligent Robots and Systems*. IEEE, 664–669.
- [28] Rochelle S Newman and Peter W Jusczyk. 1996. The cocktail party effect in infants. *Perception & Psychophysics* 58, 8 (1996), 1145–1156.
- [29] Aditya Arie Nugraha, Antoine Liutkus, and Emmanuel Vincent. 2016. Multichannel music separation with deep neural networks. In *2016 24th European Signal Processing Conference (EUSIPCO)*. IEEE, 1748–1752.
- [30] Despoina Pavlidis, Anthony Griffin, Matthieu Puigt, and Athanasios Mouchtaris. 2013. Real-time multiple sound source localization and counting using a circular microphone array. *IEEE Transactions on Audio, Speech, and Language Processing*

- 21, 10 (2013), 2193–2206.
- [31] Irwin Pollack and James M Pickett. 1957. Cocktail party effect. *The Journal of the Acoustical Society of America* 29, 11 (1957), 1262–1262.
 - [32] Arthur N Popper, Richard R Fay, and Arthur N Popper. 2005. *Sound source localization*. Vol. 25. Springer.
 - [33] Pyroomacoustics. 2014. Room Simulation. <https://pyroomacoustics.readthedocs.io/en/pypi-release/pyroomacoustics.room.html>
 - [34] Michael Risoud, J-N Hanson, Fanny Gauvrit, C Renard, P-E Lemesre, N-X Bonne, and Christophe Vincent. 2018. Sound source localization. *European annals of otorhinolaryngology, head and neck diseases* 135, 4 (2018), 259–264.
 - [35] Olga Slizovskaia, Leo Kim, Gloria Haro, and Emilia Gomez. 2019. End-to-end sound source separation conditioned on instrument labels. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 306–310.
 - [36] Yingxiang Sun, Jiajia Chen, Chau Yuen, and Susanto Rahardja. 2017. Indoor sound source localization with probabilistic neural network. *IEEE Transactions on Industrial Electronics* 65, 8 (2017), 6403–6413.
 - [37] Efthymios Tzinis, Scott Wisdom, John R Hershey, Aren Jansen, and Daniel PW Ellis. 2020. Improving universal sound separation using sound classification. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 96–100.
 - [38] Tuomas Virtanen. 2006. *Sound source separation in monaural music signals*. Tampere University of Technology.
 - [39] DeLiang Wang and Jitong Chen. 2018. Supervised speech separation based on deep learning: An overview. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 26, 10 (2018), 1702–1726.
 - [40] Gordon Wichern, Joe Antognini, Michael Flynn, Licheng Richard Zhu, Emmett McQuinn, Dwight Crow, Ethan Manilow, and Jonathan Le Roux. 2019. WHAM!: Extending speech separation to noisy environments. *arXiv preprint arXiv:1907.01160* (2019).
 - [41] Scott Wisdom, Hakan Erdogan, Daniel PW Ellis, Romain Serizel, Nicolas Turpault, Eduardo Fonseca, Justin Salamon, Prem Seetharaman, and John R Hershey. 2021. What's all the fuss about free universal sound separation data?. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 186–190.
 - [42] Robert Xiao, Greg Lew, James Marsanico, Divya Hariharan, Scott Hudson, and Chris Harrison. 2014. Toffee: enabling ad hoc, around-device interaction with acoustic time-of-arrival correlation. In *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services*. 67–76.
 - [43] Nelson Yalta, Kazuhiro Nakadai, and Tetsuya Ogata. 2017. Sound source localization using deep learning models. *Journal of Robotics and Mechatronics* 29, 1 (2017), 37–48.
 - [44] Takuya Yoshioka, Hakan Erdogan, Zhuo Chen, and Fil Alleva. 2018. Multi-microphone neural speech separation for far-field multi-talker speech recognition. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 5739–5743.
 - [45] Neil Zeghidour and David Grangier. 2021. Wavesplit: End-to-end speech separation by speaker clustering. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 29 (2021), 2840–2849.
 - [46] Cha Zhang, Dinei Florêncio, Demba E Ba, and Zhengyou Zhang. 2008. Maximum likelihood sound source localization and beamforming for directional microphone arrays in distributed meetings. *IEEE Transactions on Multimedia* 10, 3 (2008), 538–548.
 - [47] Hang Zhao, Chuang Gan, Wei-Chiu Ma, and Antonio Torralba. 2019. The sound of motions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 1735–1744.
 - [48] Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. 2018. The sound of pixels. In *Proceedings of the European conference on computer vision (ECCV)*. 570–586.
 - [49] ZM Zhong, J Chen, P Zhong, and JB Wu. 2006. Application of the blind source separation method to feature extraction of machine sound signals. *The International Journal of Advanced Manufacturing Technology* 28, 9 (2006), 855–862.