Acoustic Camera

Smitha B.[1] , Aparna V.[2] , Aravind H.[3] , Aravind K S.[4] , Gayathi A.[5] , Madhavanunni A.[6] , Gayathri M.[7]
[1]Electrical&Electronics Engineering NSS College of Engineering Palakkad, Kerala, India bsmithas@nssce.ac.in
[2]Electrical&Electronics Engineering NSS College of Engineering Palakkad, Kerala, India aparnavaloor512@gmail.com
[3]Electrical&Electronics Engineering NSS College of Engineering Palakkad, Kerala, India aravind532000@gmail.com
[4]Electrical&Electronics Engineering NSS College of Engineering Palakkad, Kerala, India aravindks2311@gmail.com
[5]Electrical&Electronics EngineeringNSS College of Engineerin Palakkad, Kerala, India gayu.sivan99@gmail.com
[6]Electrical Engineering IIT Palakkad, Kerala, India 121813001@smail.iitpkd.ac.in
[7]Electrical Engineering IIT Palakkad, Kerala, India 121814001@smail.iitpkd.ac.in

Abstract

Acoustic imaging translates sound into visual 2D representations, but its effectiveness diminishes in noisy environments. To overcome this challenge, microphone arrays can capture sound signals with reduced noise, enhancing the clarity of the resulting images. This approach falls within the realm of array signal processing. This project utilizes a 16-node microphone array system for acoustic imaging, focusing on sound source localization as a key application. Sound source data is analyzed to pinpoint and identify their locations, a critical function for real-time scenarios like rescue missions and military operations, where safety is paramount. The use of Arduino-based MATLAB facilitates sound source detection and localization, aiming to minimize localization errors. The process involves estimating the direction of arrival (DOA) and phase difference of arrival (PDOA) at the microphone sound sensors to compute localization parameters. Arduino Mega collects data from the microphone sound sensors, which are then analyzed in MATLAB. Subsequently, the data is processed in Python to generate the results, culminating in the creation of a thermal image.

Keywords—*Acoustic imaging, Microphone arrays, sound source localization, beamforming technique.*

I. INTRODUCTION

Microphone arrays are capable of capturing noise-free sound signals, making them an integral part of array signal processing. This technology provides a robust method for obtaining clean recordings amidst noisy backgrounds. In our project, we employ a 16-LM393-microphone array system for acoustic signal conversion. To counteract noise in challenging environments, we utilize the delay-and-sum technique, effectively amplifying signals. This process enhances sound intensity, facilitating its visualization. We leverage both array and beamformer techniques to map sound intensity and its corresponding visual representation. Ultimately, our project culminates in generating a thermal image, which serves as an acoustic map depicting signal intensities captured by the microphone array. This mapping proves invaluable across diverse domains by aiding in the differentiation of sound intensities. Our experiment results in a thermal image which is the acoustic mapping of intensities obtained from the microphone by which we can differentiate sound intensities which is useful in different domains. In addition to this, we also found out the PDOA (Phase difference of arrival), angle of arrival, and a map showing position information of microphones.

The distribution of amplitude of the sound in a plane could be recorded and reconstructed is done in the process of acoustic imaging. The important applications are in the medical field and underwater imaging. As the speed of propagation is low in air acoustic imaging is susceptible to more noise and chance of echo, but it is more effective in water and in the human body. The possible solution for acoustic imaging in air is yet to be explored.

Acoustic detection and localization is a hugely explored area that uses microphone arrays. In this work, sound is captured using arrays arranged in a pattern. The work is implemented in a hardware-based platform with the integration of algorithms. The sound signals are captured using microphone arrays arranged in patterns. The concept of beamforming is used here for noise cancellation. While sound waves are traveling in the same direction and of the same frequency they interfere with each other. The interference can be destructive or constructive. They cancel each other out if it is out of phase known as destructive. Their amplitudes add up if it is in phase known as constructive. The pattern in constructive fashion is known as a beam pattern. A pattern of constructive and destructive interference is made by sending multiple signals in identical forms at different times [1]. An array of sensors receive signals at different delays and if the signal is directed accordingly, it can be combined to form the expected pattern of radiation. Beamforming [2] explains that in microphone arrays, each microphone receives signals with varying time delays, allowing for the determination of the magnitude and direction of sound sources based on the time gap between the sound event and the received signal. These values are utilized to create a visual representation of the sound source. With the decreasing cost and increasing accessibility of microphone array technologies, there is a rising interest in leveraging these concepts to capture contextualized audio events for developing context-aware applications. Recent advancements in processor technology have made it easier to incorporate small arrays of microphone into consumer devices such as personal gaming devices and smartphones.

The LOUD attempt which is initiated by the MIT Computer Science and Artificial Intelligence Laboratory (CSAIL) [3], [4] resulted in an array made up of 1020 microphones. The rectangular array had a uniform arrangement of microphones which were placed at a distance of 3.0 cm apart on a panel that was about 180 cm wide and 50 cm high. These 1020 nodes were distributed across 510 PCBs, each consisting of two microphones, stereo Analog-to-Digital Converter (ADC), and a small cooling component. In doing this, the ADC sampled analog inputs at the rate of 16 kHz producing a channel of twenty four bits for sixteen serial data. The PCB modules were interlocked together like LEGO bricks such that there was one chain with sixteen modules feeding into one input on connector board with time division multiplexing as shown in figure below. Four connector boards were used, each supporting eight chains of these PCB's. This configuration enabled the total data rate to be increased to 393 Mbits/sec. To overcome this high bandwidth requirement, Raw ISA-based custom parallel processor was designed and deployed by researchers involved in LOUD project.The LOUD project concentrated primarily on assessing automated speech recognition (ASR) performance.To this end, it showed remarkable improvements in word error rates (WERs) against standalone far-field microphones when used for speech recognition tasks. Major drops in normal conditions (89.6% WER drop) and noisy environments (87.2% WER drop) were observed, respectively. The advancement of technology and the availability of cheaper microphone array technologies have led to increased interest in using them for capturing context-aware audio events for developing context-aware applications. Additionally, technological progress has made it easier to incorporate small-scale microphone arrays into consumer electronic devices such as smartphones and personal gaming devices. Most of the currently trending phones have at least two mics that are used for noise cancellation purposes. For example, the Lumia 925 [5] uses three microphones which allows enhanced microphone array-based applications like automatic voice tracking. Many companies have created commercial products for audio beamforming. For instance, Microsoft's Kinect [6] sensor is equipped with a linear microphone array designed to enhance sound from one direction while minimizing sound from other directions through beamforming technique used to improve sounds emanating from a particular direction while attenuate those coming from other locations too far from it . The Microsoft Kinect microphone array consists of four microphone capsules with each channel capable of handling 16-bit audio with a sampling rate of 16 kHz. With this feature, the device effectively suppresses ambient noise leading to the control option within games being carried out through recognizing the voice. PC-based beamforming systems using sound acquisition arrays of tens to over a hundred elements have been developed by companies such as Acoustic Camera [7]. CXS000[8] unified conference station which is an audio conferencing tool introduced by Polycom and Microsoft that has microphone array technology.

Numerous studies have explored Phase Difference of Arrival (PDOA) techniques for spectrum monitoring in radio receiver technology. These techniques leverage compact computing power to precisely calculate the timing and waveform comparisons based on source points. While single acoustic event sources can typically be detected with high accuracy, the challenge lies in identifying acoustic signals amidst overlapping noise. Indeed, the issue of overlapping signals has garnered significant attention in speech processing. As a result, PDOA methods are increasingly gaining popularity. In my research area, I am particularly interested in ensuring that PDOA techniques are not only effective in theory but also practical in hardware implementation, enabling precise determination of sound locations. To determine the source of sound, it's crucial to ascertain both the Direction of Arrival (DOA) and the Phase Difference of Arrival (PDOA). DOA estimation typically involves utilizing a group of acoustic microphone sensors. Various types of microphones employ different methods to convert sound energy.

II. METHODOLOGY

The project employs the LM393 microphone sensor. This sensor facilitates the straightforward detection of sound intensity and finds common applications in security, switching, and monitoring systems. Its accuracy is adjustable to suit various usage scenarios. The module utilizes a microphone to provide input to an amplifier, peak detector, and buffer, enhancing its functionality. The project employs a set of microphone arrays to capture signals, which are then analyzed using cross-correlation functions and a spatial gradient approach based on time-delay methods. Direction of arrival (DOA) estimation involves utilizing acoustic microphone sensors arranged in arrays. Signals received by these sensors, albeit identical, arrive at slightly different times due to their spatial separation. Beamforming techniques are applied to enhance sound localization accuracy. This endeavor to improve efficiency in sound localization represents a promising avenue for future research. The Arduino Mega 2560 is used here to process the sound signals captured by the microphone array, We use a microphone array of 16 microphones which captures the sound signal. Now the Arduino output is processed in MATLAB. In MATLAB we perform the operation of Beamforming. The coding will be done in MATLAB and the instructions are communicated to Arduino module through serial communication and then to the system.

Fig 1 Arduino Interfacing

The output from MATLAB is processed in Python for thermal image formation. Acoular, a Python-based framework, specializes in acoustic beamforming for applications in acoustic testing. It focuses on processing and analyzing multichannel data captured by microphone arrays to create mappings of sound source distributions. These maps, known as acoustic photographs, enable the localization and characterization of sources of interest based on their spectra. It's a Python library designed for handling multichannel data, typically from acoustic measurements using microphone arrays. The primary objective of the processing is to generate maps depicting acoustic sources. Some key features of the framework are it encompasses various beamforming algorithms and advanced deconvolution techniques, operates in both time and frequency domains, enables 3D mapping, caters to stationary and moving targets, offers both scripting and graphical user interface support, incorporates intelligent caching and parallel computing capabilities, and is easily extendible and well-documented. In essence, it's akin to capturing an acoustic photograph of sound sources.

Figure 2 illustrates a pictorial diagram of the project, demonstrating the placement of a sound source alongside the utilization of a microphone array to capture sound signals. A microphone array comprises multiple microphones operating sequentially to achieve its function.

Microphone arrays generate directional responses or beam patterns based on their configuration. They find diverse applications including speech recognition systems, hearing aids, and military uses such as tracking the trajectory of bullets. The sound sensor module offers a simple means of detecting sound and is commonly employed for measuring sound intensity. Its applications range from security and switching to monitoring systems. Upon detecting sound, the module generates an output signal voltage, which is transmitted to a microcontroller for further processing. Now the output from the microphone arrays is subjected to Beamforming processes using MATLAB.

Beamforming is a spatial filtering method that aims to maximize the output power towards the direction of the sound source. During transmission, multiple identical signals are emitted with slight time differences, leading to constructive and destructive interference patterns. This technique allows preferential directionality towards desired targets. In reception, a set of sensors captures the signal with varying delays, which are then combined to selectively receive the expected radiation pattern. After beamforming the output is processed again in MATLAB, processed in Python and a thermal image is obtained as shown in fig.2.

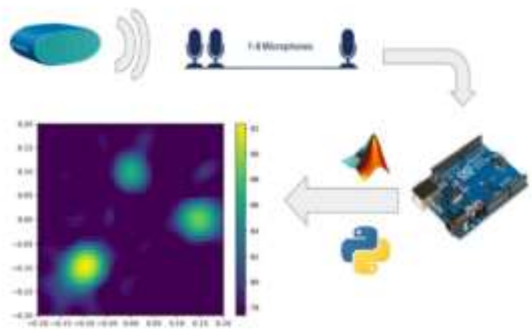

Fig 2 Pictorial diagram

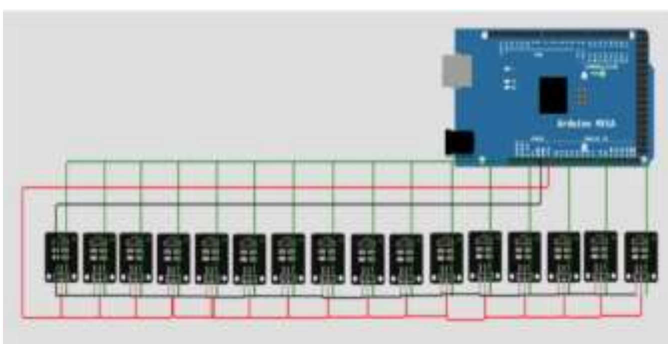

Fig 3 Circuit diagram

A. Circuit Diagram

Here, we have used 16 microphones as a microphone array to collect the sound signals and connect them as shown in fig. 3. The 16 microphones used here to increase the accuracy of the result. The result is then processed in MATLAB and fed to Python for thermal image formation.

Intensity information has been taken using 16 microphones. Then phase shift, phase difference, angle of arrival is evaluated. When phase difference is evaluated, it is checked that whether phase difference is zero or the actual value of phase difference is taken and then a final value is decided. Using the phase difference and angle of arrival details we get to determine the position information. It contains channel and frequency samples. Again, beamformed for better results. The position output is processed in Python using Acoular Library and processed and a thermal image formation occurs.

B. MICROPHONE SIGNAL TO TIME DATA CONVERSION

The intensity is taken from microphones in a series of cycles.51198 cycles of intensity are taken for the image formation. 10cycles of 4 distinct microphone intensity are taken for phase shift evaluations.

The phase shift is calculated as shown below.

Let Vo, V_1... V_n = Intensity of mics at first cycle.

Let $X1 = A0 \,.\, A1(\text{dot product})$

number of samples for successful processing of the intensities further in the implementation.

$Y = (\text{norm})V0 * V1(\text{cross product})$

$\theta = \cos\left(\frac{x}{y}\right) X \left(\frac{180}{\pi}\right)$, similarly, up to V_n.

The phase sequence is calculated as shown below.

Let A_0, A_l,..... = Average of input intensity from mics in 4 adjacent cycles,

$X1 = A0 \,.\, A1(\text{dot product})$

$Y = (\text{norm})A0 * (\text{norm})A1(\text{Cross product})$

$\theta 1 = \cos\left(\frac{x}{y}\right) X \left(\frac{180}{\pi}\right)$,

$X2 = A0. A2\ (\text{dot product})$,

$Y = (\text{norm})A0\ X\ (\text{norm})A2(\text{Cross product})$,

$\theta 2 = \cos\left(\frac{x}{y}\right) X \left(\frac{180}{\pi}\right)$,

$\text{phase sequence} = (\theta 1 + \theta 2)/2$, similarly, up to Vn.

The angle of arrival is calculated as shown below,

$\text{Angle of Arrival}_{12} = (\text{sin of phase difference x } \lambda) \times (2 * \pi * l)$ $\text{Angle of Arrival}_{23} = (\text{sin of phase difference x } \lambda) \times (2 * \pi * l)$ where, $\lambda = (\frac{c}{l})$ (source wavelength in metres),

C= 343 m/s (speed), f=1000Hz, l=0.04 m.

= (Angle of Arrival_{12}+ Angle of Arrival_{23}) / 2

The phase difference is determined as shown below in fig. 4.The input is taken from four distinct microphones and is evaluated using an if loop in the following pattern as shown, if any of the value taken by one or microphone is less than or equal to zero, then that particular microphone will be eliminated and the average of the other left microphones will be taken and if the value taken by all the four microphones is zero then the phase difference is taken as 90 degrees.

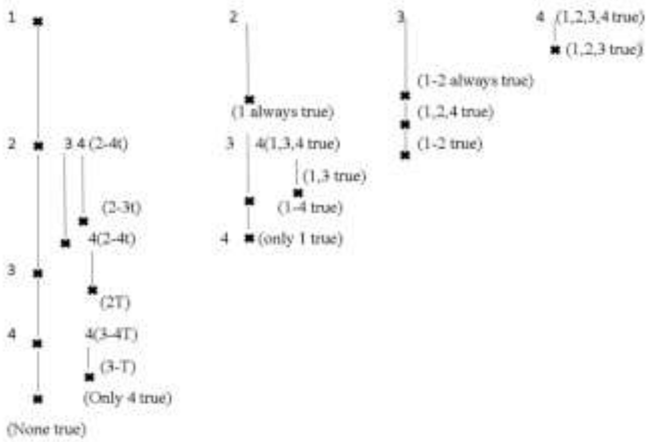

Fig 4 Phase difference flowchart

C. Simulation

Simulation is done in MATLAB. Where four analog inputs are taken as microphones input which are interfaced to the system code using Arduino. With regular time intervals the intensities from four microphones which is graphically indicated as four analog input and it is correspondingly converted to digital inputs before forming the arrays. With the random analog input details the phase sequence is calculated and also calculated the phase difference, and the angle of arrival(AOA).

D. Simulation Results

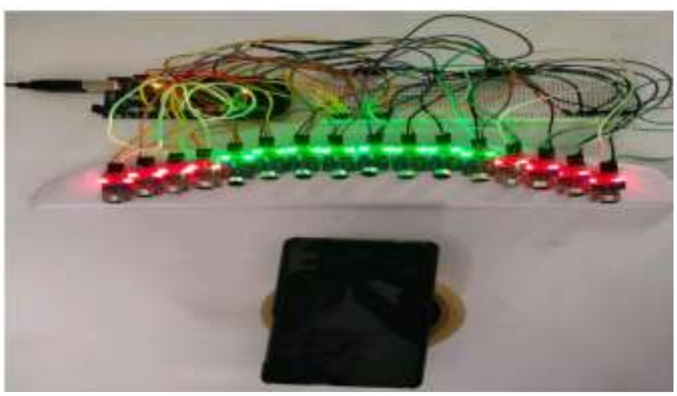

Fig 5 Hardware

On scrutinizing checking of the compatibility with subsequently increasing number of microphones that is analog inputs which is given as input to Arduino we got the maximum resolution output thus we took 2-hour input that is 5000 input intensities of each 16 microphones as in fig.5.

With 16 inputs we got a single-colored image. Thus, we replicated inputs in multiples of 16 and finally got an acceptable image with 64 inputs. thus, we finalized with 64 inputs. Then took 5-hour input that is 51198 inputs from each microphone of 16 microphones and created an array of 16*51198 array. This is further replicated to 64 microphone inputs that is 64 column inputs, and arrived at an array of 64*51198 array. This constitutes the intensity information from the microphone Along with the same we randomly took 4 microphone phase shift is calculated and subsequently phase sequence is found out.

Waves with matching cycles are termed "in-phase waves," whereas waves with phase disparities and no overlap are referred to as "out-of-phase waves." If the phase difference is greater than zero, we accept the actual phase difference otherwise it's taken as 90 degrees. The Angle of Arrival (AOA) of a pulse indicates the location from which the signal, such as radio, optical, or acoustic, is obtained. It is determined by measuring the Time Difference of Arrival (TDOA) in between specific elements of an array. Typically, this TDOA measurement involves comparing the received phase difference at each array element. This process can be conceptualized as the reverse of beamforming. In beamforming, the signal received by each element undergoes manipulation to direct the gain of the array. Conversely, in Angle of Arrival (AOA) estimation, the arrival delay at each element is directly assessed and transformed into an AOA value.. As the position of microphones changes the Angle of arrival correspondingly changes. The angle of arrival along with phase difference constitutes the position information that is required for further beamforming and imaging actions done in the coming stage. Thus, we get the intensity information and position.

Thus, we get the intensity information and position information as output from MATLAB with Arduino add on as the first phase of our implementation. In second phase we take both the information from MATLAB which is given as input to python. of which intensity information processing is done first. Here the input information in excel is converted to HDF5 format where the input information which is an array of 64*51198 is now split into three components were 1) Number of channels=64, 2)Number of samples=51198 3)Sample frequency=51198. Here the sample frequency should match the

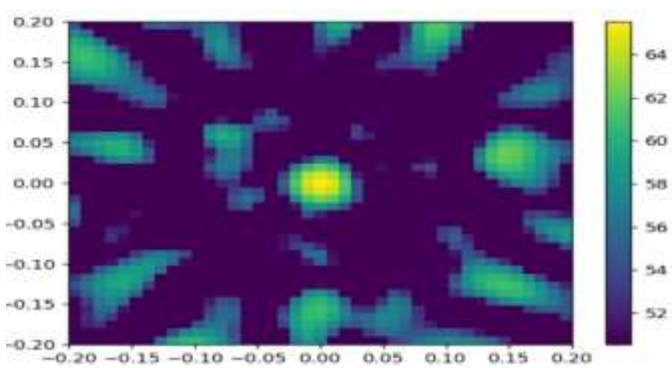

Fig 6 Image output number of samples for successful processing of the intensities further in the implementation.

Then the time domain intensities are converted to frequency domain where the output is a power spectrum that is Cross Spectral Matrix using power spectra class of acoular library. Here Fast Fourier transform is done with hanning window of block size 128.The obtained matrix is of three dimension with x=65, y=64, and z=64. After that a rectangular grid is formed with x axis and y-axis ranging from -0.2 to 0.2 and a random value for z -axis with a 0.01 per value increment. Now the position information is imported from MATLAB and hardware positioning values are combined. Then using acoular library and that is visualized using Matplotlib.

Then the Steering Vector class is utilized to form a steering vector. Thus, the combined position output from MATLAB and hardware is combined in the steering vector and beamforming is done correspondingly. In this context, we aim to calculate the sum of all Fast Fourier Transform (FFT) frequency lines within the 3rd octave band centered at 8000 Hz. This involves generating maps for each frequency within this band and then aggregating them to simulate the outcome of a third-octave filter for that specific band. The outcome represents the mean square sound pressure contribution at the central location of the microphone array. In acoustics, results are typically expressed in terms of sound pressure levels (SPL). Acoular provides a utility function called "Lp" to compute SPL levels from mean square values. Then logarithm of this output is taken to form the thermal map which shows the sound intensity variation. The current output received is a thermal map that matches the dimensions of the grid. Any zero results in the map will be adjusted to a level of -350 dB instead of being represented as infinity. Now, let's visualize the map as a thermal image, ensuring a dynamic range of 15 dB between the highest and lowest ends of the color scale [7].

III. CONCLUSION

We tried out with different arrangement of microphones that is in concave circular semi-circular linear and as arrays. The conclusion was that position gives the most accurate results. While coming to number of microphones we first tried out with 4 which we increased sequentially and fixed at sixteen because it is the maximum capacity of Arduino MEGA. And it is the maximum multiple of 4 which can be correspondingly replicated to obtain the final image. Thus, we conclude that we take input intensities from microphone and convert them to image corresponding showing the intensity differences. And this is just a prototype of how an acoustic camera works and automated to give image when sound is given.

The main limitation is the availability of high sensitivity microphones module, and to get clear thermal image. Then the next limitation is to reduce the noise in great accuracy if placed in a noisy environment, Other can be to obtain direction of arrival many arrays position will not give the output so to arrive at a compatible and accurate position or configuration of microphone is of great difficulty.

The acoustic camera finds numerous applications, primarily centered around noise reduction. In biomedical contexts, it can be integrated into stethoscopes, mitigating the risks associated with ultrasound while reducing time and expenses in obtaining results. Moreover, the acoustic camera is extensively utilized to refine the noise emission properties of various vehicles (e.g., four wheelers, aircraft, trains) and installations like air turbines and vast machinery used in boring operations. It not only measures external noise emissions but also enhances cabin comfort in vehicles. Spherical acoustic cameras are preferred for such tasks due to their ability to localize sound sources in all directions. Additionally, they aid in troubleshooting mechanical faults in machines and mechanical parts by comparing sound mappings of properly functional machines with those of damaged ones. Acoustic cameras are also valuable in studying noise levels inside passenger carriages during train operations. Placing the camera near train tracks offers an alternative perspective on the noise experienced inside the train. Furthermore, an external setup can investigate the squealing of train wheels caused by track curves.

A key objective is to develop a system capable of scaling this technology for application in a stethoscope. This involves miniaturizing the components and employing high-sensitivity microphones along with Field Programmable Gate Array (FPGA) technology. This approach aims to reduce the time required to capture input intensities from human internal organs, enabling the imaging of sound intensity differences on a screen for diagnostic purposes. Thus, we can reduce some of the cases referred to taking ultrasound and this can be an added advantage to both the medical practitioner and the patient. It also reduces the time for obtaining the results. Employing a Neural Networks-Based approach could enhance the effectiveness of analyzing acoustic signal propagation, especially when dealing with extensive datasets and aiming to discern phase sequence, phase differences, and sound intensities. By leveraging this approach, algorithms can be developed to learn calculations of Direction of Arrival (DOA) and Time Difference of Arrival (TDOA), particularly beneficial for designing sophisticated algorithms for Sound Source Localization (SSL) in robotics. Furthermore, this technology could be implemented in hospital rooms to monitor patients' cough histories, particularly those suffering from respiratory diseases. Additionally, utilizing a specialized microphone array capable of detecting heart rate irregularities could further enhance diagnostic capabilities.

REFERENCES

[1] Nguyen, H. (2020). Introduction to the Acoustic Camera. Metropolia University of Applied Sciences, Bachelor of Engineering Environmental Engineering

[2] Dobler, D., & Heilmann, G. (2005, July). Perspective of the acoustic camera. In INTER-NOISE and NOISE-CON Congress and Conference Proceedings (Vol. 2005, No. 6, pp. 2491-2498). Institute of Noise Control Engineering.

[3] Weinstein, E., Steele, K., Agarwal, A., & Glass, J. (2004). A 1020-node modular microphone array and beamformer for intelligent computing spaces.

[4] Cutler, R., Rui, Y., Gupta, A., Cadiz, J. J., Tashev, I., He, L. W.. ... & Silverberg. S. (2002, December). Distributed meetings: A meeting capture and broadcasting system. In Proceedings of the tenth ACM international conference on Multimedia (pp. 503-512).

[5] Rakesh, P., Priyanka, S. S., & Kumar, T. K. (2017, March). Performance evaluation of beamforming techniques for speech enhancement. In 2017 Fourth International Conference on Signal Processing, Communication and Networking (ICSCN) (pp. 1-5). IEEE.

[6] Ali, E., Ismail, M., Nordin, R., & Abdulah, N. F. (2017). Beamforming techniques for massive MIMO systems in 5G: overview, classification, and trends for future research. Frontiers of Information Technology & Electronic Engineering, 18(6), 753- 772.

[7] Priyanka, S. S. (2017, April). A review on adaptive beamforming techniques for speech enhancement. In 2017 Innovations in Power and Advanced Computing Technologies (i-PACT) (pp. 1-6). IEEE.

[8] Luijten, B., Cohen, R., de Bruijn, F. J., Schmeitz, H. A., Mischi, M., Eldar, Y. C., & van Sloun, R. J. (2020). Adaptive ultrasound beamforming using deep learning. IEEE Transactions on Medical Imaging, 39(12), 3967-3978.

[9] Zhao, J., & Ritz, C. (2020, December). Semi-Adaptive Beamforming for Co-Prime Circular Microphone Arrays. In 2020 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC) (pp. 426-431). IEEE.

[10] Stroobandt, D., Verplaetse, P., & Van Campenhout, J. (2000). Generating synthetic benchmark circuits for evaluating CAD tools. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 19(9), 1011-1022.