BeBeC-2024-D06

HARDWARE DESIGN AND SOFTWARE IMPLEMENTATION OF A PORTABLE GPU-BASED 3D ACOUSTIC CAMERA FOR INDUSTRIAL NOISE MANAGEMENT

Loic Boileau[1], Joël Lemay[1], Kevin Rouard[2], Julien St-Jacques[2], Olivier Doutres[2], Thomas Padois[3], Frank Sgard[3], Hugues Nélisse[3], François Grondin[1], Alain Berry[1] and Nicolas Quaegebeur[1]

[1]Université de Sherbrooke (UdeS),
2500 Bd de l'Université, J1K2R1, Sherbrooke (Québec), Canada
[2]École de technologie supérieur (ÉTS),
[3]Institut de recherche Robert-Sauvé en santé et en sécurité du travail (IRSST)

ABSTRACT

The need for advanced noise source localization technologies in noisy and confined workplaces has grown due to the rising instances of noise-induced deafness cases among industrial workers. This project delineates the hardware design and software implementation of a portable, GPU-based 3D acoustic camera to identify and characterize noise sources within enclosed work environments.

The proposed solution combines a spherical microphone array (64 MEMS digital microphones) and state-of-the-art signal processing techniques on a portable GPU platform. Overcoming the challenge of high microphone count and spacing is achieved through time-division multiplexing (TDM) and a modified Integrated Interchip Sound (I^2S) protocol, extending the classical 50 mm limit between adjacent microphones by up to 400 mm.

This method allows a simplified implementation of a cost-effective extended 3D arrays, as well as an easy integration within the existing portable GPU platforms available on the market. Initial validation on a 315 mm radius open sphere, combined with a 360-degree depth camera, achieves a 30 Hz refresh rate over a 150,000-pixel grid. The system's precise noise source localization and real-time imaging represent significant advancements in workplace noise assessment, ultimately contributing to the prevention of noise-induced hearing loss.

1 INTRODUCTION

In Québec (Canada), 86% of occupational disease claims in 2021 are linked to hearing-related issues [1]. Yearly, around 300 000 manual workers are exposed to harmful noise levels [2]. Québec modified its regulations to mitigate continuous noise exposure, aligning its legal standards with those of other provinces. However, the most effective approach to reducing noise levels is by addressing them at their source. It is then essential to be able to identify and locate spatially the sources contributing to workplace noise.

Acoustic antenna arrays offer a promising approach to identifying and classifying sound sources [3]. These arrays capture spatial information through multiple microphones arranged in different shapes like a sphere, providing a comprehensive overview of surrounding sources in a single measurement [4, 5, 6, 7]. Coupling such arrays with a depth camera gives access to the source position by obtaining the distance information of each pixel—this combination is referred to as an acoustic camera [8, 9]. This has been done multiple times, but mainly with 2D cameras due to the complexity and cost of obtaining a 3D depth sensor [10].

Existing commercial acoustic cameras lack accessible and affordable real-time imaging possibilities with integrated processing systems, a solution essential for industrial applications. This is attributed to both physical constraints and technical challenges, such as inter-microphone distances (resulting from the limitations of the I^{2}S interface when dealing with a large number of microphones) and the computational complexity of localization algorithms, as highlighted in [8]. Moreover, imaging industrial noise sources poses additional challenges, as industrial environments include many reflective surfaces which generate phantom sources, underscoring the necessity for precise distance estimation.

The acoustic camera described in this paper aims to address those physical limitations by using time-division multiplexing (TDM) and a state-of-the-art mobile device with a graphical processing unit (GPU) for fast and accurate sound source localization [11, 12]. The system uses a 64 spherical digital MEMS microphone array and a 3D depth camera [13]. The result is a fully autonomous and modular “open” spherical antenna with a radius of 315 mm which can generate 3D acoustic images with a frame rate of 20 frames per second with a visually acceptable pixel grid size (150 528 pixels, 224 x 672 of width and height) and share the beamformed image through a low latency network protocol.

The device comprises four principal modules: an audio acquisition module, which collects audio data at a rate between 32 and 48 kHz; a video acquisition module to obtain precise distance information; an imaging algorithm implemented on GPU; and a client/server user interface, to ease visualization of the acoustic image. The following section provides a comprehensive overview of those modules, with the calculations for determining the maximal distances between each MEMS microphone.

2 METHODS

2.1 Audio data acquisition

Achieving a compact 315 mm antenna requires small microphones, a feature made feasible with the introduction of MEMS microphones. Digital MEMS offer an improved solution for portable antennas, given their capability to transmit encoded digital data through established protocols like I^2S [14].

The I^2S interface allows for the overlay of two digital audio signals using a high-speed clock. With a 64 MEMS microphones antenna, it requires 32 I^2S ports, leading to increased complexity and costs. To address this issue, the time-division multiplexing (TDM) interface operates through the I^2S bus, enabling the encoding of up to 16 audio channels over a single Serial Data line (SD). This is achieved using a Word Select signal (WS) alongside a high-frequency clock, determining which microphone can write its data without overlap. Each microphone triggers its Word Select Output (WSO) when finished writing, signalling the next microphone. This approach reduces the required number of I^2S ports to 4 slots, significantly reducing costs for the antenna. Still, a custom circuit needs to encode the microphone signals in TDM, potentially with Field Programmable Gate Arrays (FPGAs) or dedicated Digital Signal Processing (DSP) codec, thereby increasing costs.

Currently, the ICS-52000 from Invensense[1] is the only digital MEMS microphone available with a 16-channel TDM interface without requiring a custom encoder [15]. According to its specifications, due to propagation delays and clock signal impedance, the maximum distance between microphones is limited to 50 mm. Daisy-chaining 16 x ICS-52000 units within this inter-microphone distance is hardly feasible for a 315 mm diameter spherical microphone array. To address this limitation, organizing the microphones in a parallel configuration could reduce the distance between the furthest microphone and the computer. However, this necessitates synchronizing the signals to prevent simultaneous writing.

Figure 1 illustrates theoretical calculations for signal propagation delay in both the standard daisy-chain configuration and a 4x4 parallel configuration chosen for the antenna described in this paper. This configuration adds a D-type flip-flop in-between delaying the signal one bit permitting to remove the delays caused by the left-side loop in Fig. 1 (b). This is adjusted in the acquisition driver of the on-board computer, effectively modifying the I^2S protocol. Assuming a signal speed of *0.57c*, where *c* is the speed of light, the maximum length for each branch of 4 microphones in the 4×4 configuration is approximately 1800 mm. Therefore, with a safety factor of 10% considered, it is reasonable to expect the ICS-52000 to function effectively with an inter-microphone distance of up to 400 mm.

[1] Note: the optical MEMS microphone SMB100 from sensiBel also offers TDM but is limited to 8 channels

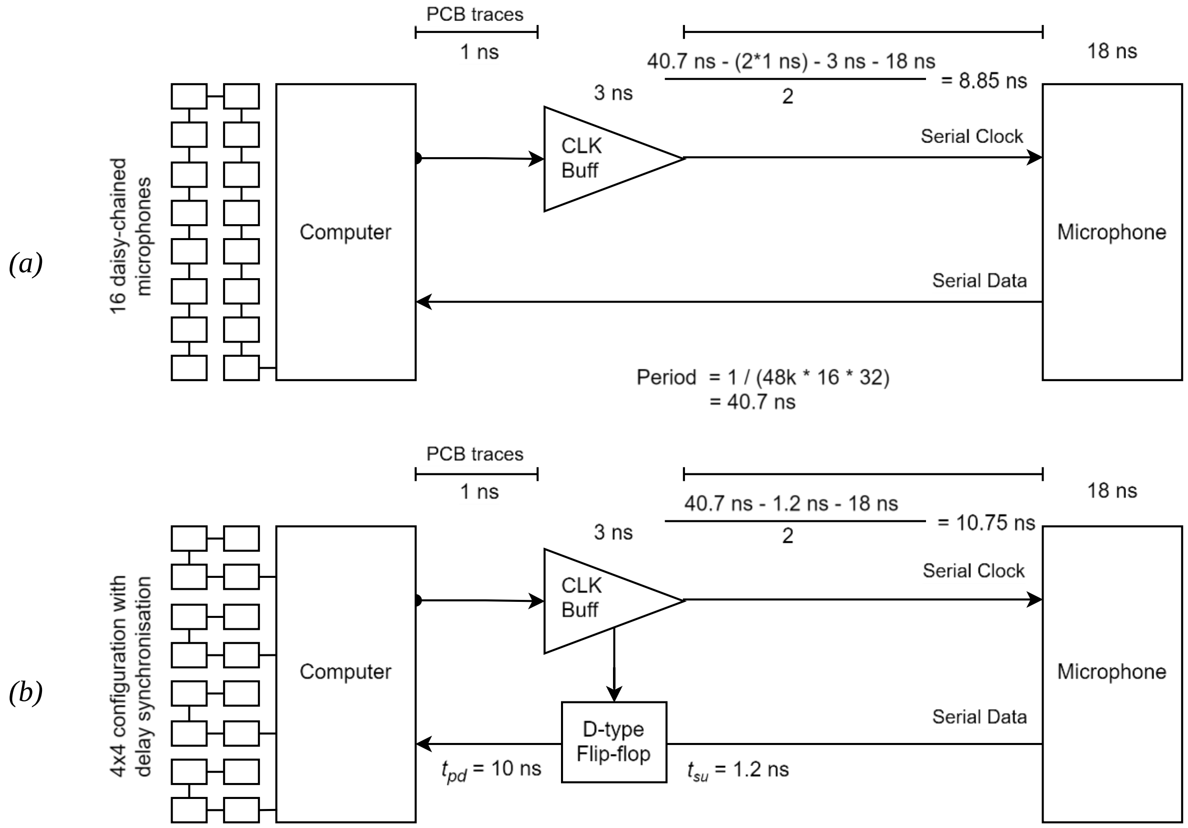
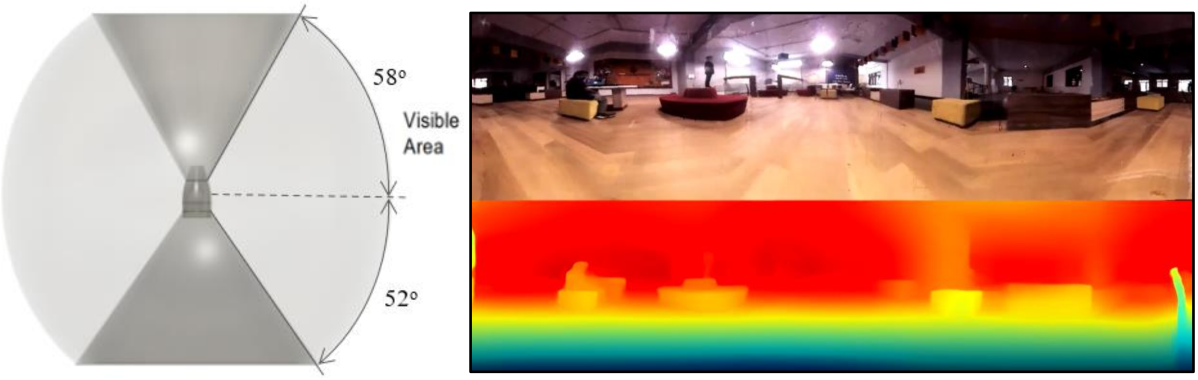


Fig. 1. Maximal delay for signal to reach an ICS-52000 microphone in (a) 16 daisy-chain configuration and (b) with a synchronisation flip-flop for a 4x4 configuration with a frequency of 48 kHz and each numeric data encoded on 32 bits.

2.2 Video data acquisition

The focusing distance for beamforming must correspond to the source-array distance, requiring this distance to be measured. Applying a focusing distance adapted to the actual source-array distance reduces the distortion of the acoustic image [9] and allows weighting of near-field sources. As mentioned earlier, the distance estimation requires a 3D depth camera, which is now readily accessible with various available options.

Companies such as StereoStitch, Vuze, and DreamVu offer ready-made solutions for capturing 360-degree depth images. StereoStitch provides software that merges images from multiple cameras to produce real-time stereoscopic videos. Vuze (and many others) offer a compact 360-degree camera capable of capturing 3D and high-resolution videos, although it appears to lack real-time functionality. In contrast, DreamVu offers the PAL camera, which utilizes mirrors to generate a real-time stereoscopic 360-degree depth image using only one camera. Connected through a USB cable for data transfer, this camera relies on the GPU of the onboard computer to perform calculations for generating the depth map. It offers a panoramic field of view (FOV) of 110 degrees vertically and 360 degrees horizontally, as illustrated in Fig. 2. Its effective maximum working distance ranges from 5 to 10 meters depending on its height.



(a) *(b)*

Fig. 2. (a) PAL-USB 3D depth camera FOV and (b) an illustration showcasing its depth perception including an image positioned at the top, and below, a heatmap indicating distances (where blue signifies proximity and red signifies distance).

2.3 Algorithm implementation

Following previous studies using GPUs for real-time imaging [11, 12], an onboard platform with an integrated GPU has been selected since all the needed calculations can be performed in a time frame fast enough to be able to produce image in real-time with a total latency of 100 ms and a framerate of 20 frame per second (fps).

The localization algorithm chosen is the Generalized Cross-Correlation with Phase-Transform (GCC-PHAT) [16]. This algorithm starts by calculating the cross power spectral density (CPSD) for each microphone pair (m,n).

$$G_{p_m,p_n}(k) = P_m[k]P_n^*[k], \qquad (1)$$

where k is the discrete frequency index, $P_m[k]$ and $P_n[k]$are the discrete Fourier transform of microphones m and n pressure signals and * denotes the complex conjugate. As the signals are obtained in real-time, a Short-Time Fourier Transform is more appropriate to obtain P_m and P_n, using a window size N of 1024. The degree of overlap, window size, and zero padding can all be adjusted to enhance resolution and contrast. It is possible to obtain the approximation of the cross-correlation of each microphone pair by applying the inverse Fourier $\mathcal{F}^{-1}$ transform and the PHAT φ_{PHAT} transform to the CPSD:

$$R_{y_m y_n}[s] = \mathcal{F}^{-1}\left(\varphi_{PHAT} \cdot G_{p_m,p_n}(k)\right) = \mathcal{F}^{-1}\left(\frac{P_m[k]P_n^*[k]}{|P_m[k]||P_n^*[k]|}\right), \qquad (2)$$

where y_m and y_n are the filtered signals and $s \epsilon \{0, \ldots, N-1\}$. By having spatial data for each pixel, which corresponds to potential sources, the Euclidean distance between each microphone pair can be calculated. Then, this distance must be divided by the speed of sound to obtain the delay between these two microphones $\tau_{mn}(i)$ such as

$$\tau_{nm}(i) = \frac{|\overrightarrow{r_m}-\vec{r_i}|-|\overrightarrow{r_n}-\vec{r_i}|}{c_0}, \tag{3}$$

where $\overrightarrow{r_m}$ and $\overrightarrow{r_n}$ are the position vectors of microphones m and n, $\vec{r_i}$ is the cartesian position vector of pixel i and $c_0 = 343$ m/s is the speed of sound in air at ambient temperature. Finally, to obtain the acoustic image $I(i)$, the sum of the values obtained by indexing cross-correlation vector with the delay calculated for each pixel for all microphone pairs needs to be evaluated. This will return the relative gain for each pixel.

$$I(i) = \frac{1}{M_p}\sum_{m=1}^{M-1}\sum_{n=m+1}^{M-1} R_{y_m y_n}(\tau_{mn}(i)), \tag{4}$$

The same process was described in [17]. However, since the algorithm has a complexity of $O(M^2)$, it requires significantly more calculations due to the increased number of microphones, now totalling 64 instead of 11. In order to implement the algorithm on a GPU, it is important to break it down into kernels, which are individual functions executed within the numerous GPU cores. The on-board computer employed in this array system is the NVIDIA's Jetson Orin AGX, currently recognized as the cutting-edge solution in on-board computing featuring GPU capabilities. It uses a 2048-core NVIDIA Ampere architecture GPU with 64 Tensor Cores and a 12-core 64 Gb CPU. The CUDA library, provided by NVIDIA, is commonly used for GPU coding. It's worth mentioning that memory access within the kernels can impact performance, potentially leading to speed improvements if managed effectively. Figure 3 depicts the algorithm broken down into kernels to produce an acoustic image.

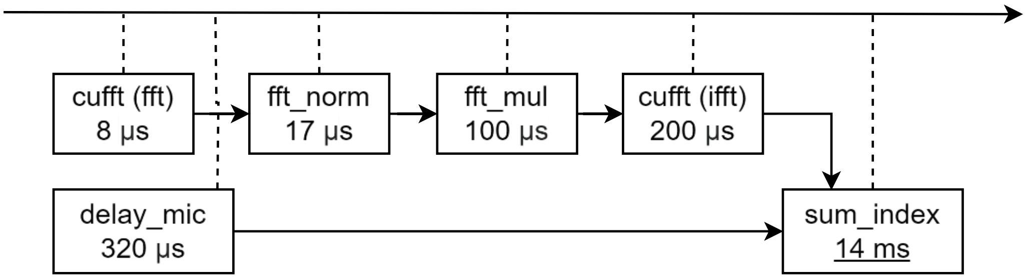


Fig. 3. GCC-PHAT algorithm separated into GPU kernels and their execution time for 64 microphones and over 150000 pixels.

The initial kernel, labelled "cufft (fft)," employs the highly optimized *cufft* sub-library in CUDA for a Real-to-Complex Fast Fourier Transform calculating P_m and P_n in Eq.1. It achieves remarkable speed, completing the transformation for 64 signals of 1024 data in less than 8 µs. It is important to note that all the kernels function with a data structure of 32-bit floats. The next kernel labelled "delay_mic," computes the Euclidean distance between each microphone and every pixel, denoted as $|\overrightarrow{r_m} - \vec{r_i}|$. The computation of τ_{mn} in Eq. (3) occurs subsequently to restrict iteration over microphone pairs. In practice, for a scenario involving 64 microphones and over 150,000 pixels, this kernel typically takes around 320 µs to complete. Following this, the third kernel, "fft_norm," computes the norm of each microphone signal $P(k)$ for subsequent application in the PHAT transform outlined in Eq. (2). Given the task of processing only 64 signals, this kernel generally executes in 17 µs. The "fft_mul" kernel follows by calculating the multiplication between the CPSD and the φ_{PHAT} filter using the "fft_norm" output. Afterwards, a Complex-to-Complex Inverse Fast Fourier Transform is done to obtain $R_{y_m y_n}(s)$, as provided by Eq. 2, also using the *cufft* sub-library. Finally, the summation of the cross-correlations of all microphone pairs for each pixel is done in the "sum_index" kernel as well as the calculations and indexing of the delays, returning the relative gain shown at Eq. (4). It is worth noticing that in the example depicted in Fig. 3, the computing time of this kernel is notably longer, measured in ms. The extended duration primarily arises from the dual summation required for its computation and the necessity to access non-coalesced memory during the indexing process. This completes the image generation process on GPU, with an estimate time of 15 ms, excluding memory transfers between CPU and GPU.

2.4 Integration

Due to the decoupling of the audio and video systems, the acquisition occurs asynchronously, resulting in audio and videos updates at different rates. To overcome this, synchronization and multithreading systems are essential for initiating image generation in real-time. This is achieved through a producer-consumer structure: the "producer" thread captures signals and stores them in a First-In First-Out (FIFO) buffer, while the "consumer" thread retrieves them as needed. This architecture allows for jitter within the on-board computer without impeding its performance. Regarding the Server/client module, it utilizes a straightforward TCP/IP transfer protocol, employing two ports: one for communication (control) and another for transferring substantial data, such as the image generated at a rate of 20 frames per second.

As previously stated, the antenna is designed in an open sphere configuration. This selection is arbitrary as both open and closed sphere configurations have been demonstrated to perform effectively with GCC, even without accounting for diffraction effects in the case of the closed sphere [18]. With 16 vertical branches, each housing 4 microphones and following the circumference, the antenna boasts a total of 4 acquisition hubs utilizing the 4×4 configuration as illustrated in Fig. 1. The microphones are connected through flex-PCBs in a 2 by 2 configuration separated by 100 mm. Despite the Jetson's provision of 6 integrated I^2S ports, only 2 are physically accessible with the provided development kit. To overcome this limitation, a custom carrier board was developed to facilitate access to the required 4 ports. The first antenna prototype was engineered with adaptability in mind, enabling easy adjustment of its microphone arrangement as required. Each microphone arm boasts numerous connection points, facilitating swift attachment of the microphones using a quick-connect mechanism. The entire antenna is crafted through 3D printing, utilizing a threaded rod for its core support and retention system, along with an aluminum tripod attachment. The microphone struts are printed using resin for structural resistance, while the remaining components are made from PLA. Figure 4 shows the acoustic camera with its 315 mm spherical antenna.

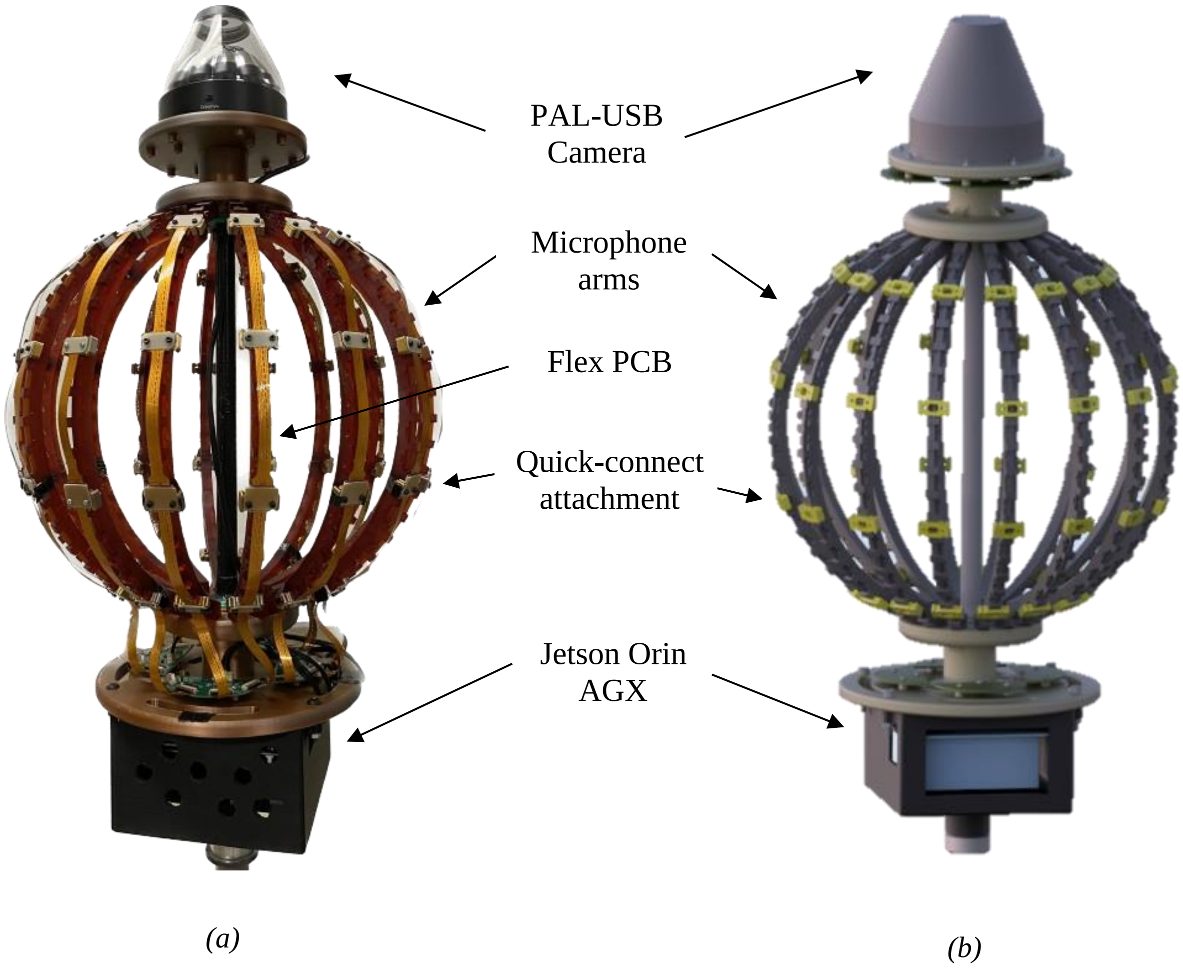


Fig. 4. (a) Actual image of the acoustic camera prototype and (b) a modeled representation. Microphones are linked via flex-PCBs and are easily attached to the antenna branches using a quick-connect mechanism.

The real-time criterion is met when the image's refresh rate is sufficiently high to create a video-like without significant latency. In this application, the target refresh rate is 20 Hz, meaning minimally 20 images should be generated per second. Regarding latency, it's quite reasonable to assume that a 100 ms delay would remain imperceptible, allowing for a margin of additional time.

In Table 1, the average time taken by each module in the creation of an acoustic image, along with the overall duration, is presented. This table covers the entirety of the process, including the acquisition of audio and video signals, image generation on GPU, and transfer to the client's user interface.

Table 1. Average duration of each module for producing an acoustic image with 64 microphones and 150 528 pixels (224x672).

Total	Audio	Video	GPU	UI transfer
30 ms	1 ms	9 ms	18 ms	2 ms

Following the table results, a refresh rate of 33 Hz is reached, demonstrating the antenna's ability to operate in real-time. To compute latency, the ping time must be added to the image production duration. For a wired antenna, the ping delay typically ranges between 10 and 40 ms. However, with wireless connections, it can extend up to 100 ms, yet still maintaining the appearance of low latency.

3 CONCLUSIONS

In summary, the goal of this project was to design a cost-effective 3D real-time acoustic camera aimed at identifying and characterizing noise sources within noisy and confined workplaces. This involved addressing technical challenges such as integrating a high number of digital MEMS microphones within a large diameter sphere, achieving a 64-microphone spherical array of 315 mm diameter. Notably, this system can produce images at a rate of 30 Hz, providing valuable insights into transient noise sources in industrial environments while maintaining the processing unit localized on the antenna, making it transportable and easily installable.

Several limitations have been identified, notably the audible noise produced by the Jetson Orin and PAL-USB camera, originating from their fan operation. This noise exhibits harmonic characteristics, which can be easily filtered or compensated for effectively. Furthermore, the PAL-USB camera currently imposes a fixed resolution for both depth and color images. While the depth resolution suffices, improving the color image quality would enhance visual clarity. Additionally, optimizing the video acquisition process could yield significant benefits. Video data received in polar coordinates necessitates conversion to Cartesian coordinates, a task currently handled by the CPU. However, transitioning this task to the GPU could reduce processing time on the video part, saving valuable milliseconds, and hence augmenting the refresh rate.

4 RÉFÉRENCES

[1] CNESST, "Statistiques annuelles, 2021 - version finale," 2022. [Online]. Available: https://www.cnesst.gouv.qc.ca/sites/default/files/documents/statistiques-annuelles_0.pdf.

[2] É. Ledoux and D. Denis, "Enquête québécoise sur des conditions de travail, d'emploi et de santé et de sécurité du travail (EQCOTESST)," *Pistes*, Vols. 13-2, 2011.

[3] L. de Santana, "Fundamentals of acoustic beamforming," *NATO Educational Notes EN-AVT-287*, vol. 4, 2017.

[4] C. Du and Q. Leclère, "Design and Evaluation of Open Spherical Microphone Arrays," in *Proceedings of ICVS*, London, England, 2017.

[5] T. Padois, O. Doutres, F. Sgard and A. Berry, "Time domain localization technique with sparsity constraint for imaging acoustic sources," *Mechanical Systems and Signal Processing,* vol. 94, pp. 85-93, 2017.

[6] T. Padois, J. St-Jacques, K. Rouard, N. Quaegebeur, F. Grondin, A. Berry, H. Nélisse, F. Sgard and O. Doutres, "Acoustic imaging with spherical microphone array and Kriging," *JASA Express Letters,* vol. 3, no. 4, 2023.

[7] N. Quaegebeur, T. Padois, P.-A. Gauthier and P. Masson, "Enhancement of time-domain acoustic imaging based on generalized cross-correlation and spatial weighting," *Mechanical Systems and Signal Processing,* vol. 75, pp. 515-524, 2016.

[8] J. Fréchette-Viens, *Développement d'une caméra acoustique pour la localisation de sources sonores,* Sherbrooke, Canada: Université de Sherbrooke, 2020.

[9] A. Meyer and D. Döbler, "Noise source localization within a car interior using 3D-microphone arrays," in *Proceedings of the BeBeC,* Berlin, Germany, 2006.

[10] B. M. Williamson, J. J. LaViola Jr, R. Sottilare and P. Garrity, "Creating a 360-Degree RGB-D Sensor System for Augmented Reality Research," in *Proceedings of the I/ITSEC,* 2018.

[11] V. P. Minotto, C. R. Jung and B. Lee, "GPU-based approaches for real-time sound source localization using the SRP-PHAT algorithm," *International Journal of High Performance Computing Applications,* vol. 27, no. 3, pp. 291-306, 2013.

[12] J. A. Belloch, A. Gonzalez, A. M. Vidal and M. Cobos, "On the performance of multi-GPU-based expert systems for acoustic localization involving massive microphone arrays," *Expert Systems with Applications,* vol. 42, no. 13, pp. 5607-5620, 2015.

[13] Z. Prime and C. Doolan, "A comparison of popular beamforming arrays," in *Proceedings of the Science Technology and Amenity,* Victor Harbor, Australia, 2013.

[14] A. Alexandridis, S. Papadakis, D. Pavlidi and A. Mouchtaris, "Development and evaluation of a digital MEMS microphone array for spatial audio," in *Proceedings of the EUSIPCO,* Budapest, Hungary, 2016.

[15] "InvenSense Announces World's First TDM Microphone With an Array of 16 Devices on a Single Bus | TDK," 2016. [Online]. Available: https://invensense.tdk.com/news-media/invensense-announces-worlds-first-tdm-microphone-with-an-array-of-16-devices-on-a-single-bus/. [Accessed 28 03 2023].

[16] C. Knapp and G. Carter, "The generalized correlation method for estimation of time delay," *IEEE Transactions on Acoustics, Speech, and Signal Processing,* vol. 24, no. 4, pp. 320-327, 1976.

[17] J. Fréchette-Viens, N. Quaegebeur and N. Atalla, "A low-latency acoustic camera for transient noise source localization," in *Proceedings on CD of the 8th BeBeC,* Berlin, Germany, 2020.

[18] K. Rouard, J. St-Jacques, F. Sgard, H. Nélisse, A. Berry, N. Quaegebeur, F. Grondin, O. Doutres and T. Padois, "Numerical comparison of acoustic imaging algorithms for a spherical microphone array," in *Proceedings of the BeBeC,* Berlin, Germany, 2022.