IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 16, 2020, accepted March 25, 2020, date of publication April 2, 2020, date of current version April 22, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2985281

Blinkies: Open Source Sound-to-Light Conversion Sensors for Large-Scale Acoustic Sensing and Applications

ROBIN SCHEIBLER, (Member, IEEE), AND NOBUTAKA ONO, (Senior Member, IEEE)
Graduate School of Systems Design, Tokyo Metropolitan University, Tokyo 191-0065, Japan

Corresponding author: Robin Scheibler (robin.scheibler@ieee.org)

This work was supported in part by the JSPS Postdoctoral Fellowship under Grant 17F17049, and in part by the SECOM Science and Technology Foundation.

ABSTRACT We propose an open source hardware platform called Blinky, a sound-to-light conversion sensor that harvests sound power at low-rate and conveniently. Blinkies are made up of a central processing unit connected to two microphones and a few light-emitting devices, are powered by a battery, and protected by a robust enclosure. Distributed in space and combined with a conventional video camera, they allow to practically sense sound power over a very large area without hassle. We give a comprehensive overview of the proposed system and its potential applications. We describe the hardware design and trade-offs made. We provide a model for the channel between sound power measurements and signal acquired by the video camera. Because each sensor is potentially affected by a different attenuation due to the channel, we propose a calibration procedure to restore the scale of the measurements. The effectiveness of the calibration is validated in an experiment. Finally, we demonstrate sound source localization using a hundred-and-one actual Blinkies in highly reverberent environment.

INDEX TERMS Blinkies, acoustic power sensors, distributed sensing, sound source localization, sound-to-light conversion.

I. INTRODUCTION

Sound is a powerful medium for sensing the world. Beyond voiced communication, sound provides very detailed information about our direct surroundings. We can, for example, distinguish a busy shopping street from a factory or a library, know when it is raining outside, or diagnose a malfunctioning washing machine. Spatial cues are fundamental to hearing in both humans and animals [1]. They come in two flavors: interaural differences in levels and time delays, that is the amplitude and the time of arrival of sound vary between the ears. Humans leverage these differences to excel at many tasks such as sound source localization and understanding. Using recordings from multiple microphones, so-called *microphone arrays*, it is possible to similarly take advantage of spatial cues for audio processing tasks [2]. Notable examples are speech enhancement via beamforming [3], [4], source separation [5], source localization [6] and tracking [7], room geometry inference [8], dereverberation [9], and acoustic scene analysis [10]. Common to most of these techniques is that they disproportionately rely on timing cues. As a result, most microphone array designs perform synchronous sampling of all the channels, for example the Pyramic [11] and LOUD [12] arrays, or that of Perrodin *et al.* [13]. Not withstanding, the design of synchronous microphone arrays is challenging. Wired microphone designs are straightforward but favor compact arrays for practical reasons, and often require specialized hardware for acquisition and processing [11]–[13].

Covering larger areas with microphones allows more sources to be close to at least one microphone, providing cleaner information about these sources. In a compact array, on the contrary, information about distant sources is tightly correlated among the microphones. To cover larger areas, wireless arrays are preferable but require taking care of synchronization and sampling frequency mismatch. In some cases, this can be achieved with an extra time alignment step [14]. Yet, wireless arrays still must contend with network

The associate editor coordinating the review of this manuscript and approving it for publication was Giovanni Pau.

bandwidth and latency issues. More fundamentally, when relying on timing cues, all wavelengths smaller than twice the inter-microphone distance will be affected by spatial aliasing [3]. For microphones spread over a large area, this drastically reduces the range of frequencies where timing cues based methods can be applied.

We recently proposed an alternative way of sensing acoustic information using Blinkies [15]. A Blinky is a compact, battery-powered microcontroller unit (MCU) equipped with at least one microphone and one light emitting diode (LED). It converts the collected sound power into a luminous signal emitted by the LED. Once distributed over space, the signals from the Blinkies can be harvested by a conventional video camera. Such a system is very easy to distribute over a large area and can easily scale up to hundreds or thousands of sensors. However, due to the frame acquisition rate of conventional video cameras, the effective sampling rate is in the 30 Hz to 60 Hz range. This makes the proposed system very different from conventional microphone arrays and suitable to tackle a different range of problems. An illustration of the system is showed in Fig. 1.

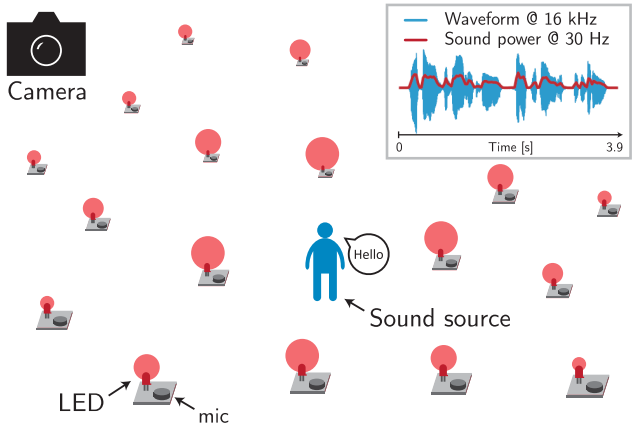

FIGURE 1. Illustration of the Blinky large-scale sensing system. The Blinkies closer to the sound source are brighter due to the larger sound power. The video camera is used to record the signals from all Blinkies simultaneously. In the upper right corner, an example of a speech signal waveform and its short term power envelope.

Such sensors seem particularly suitable for auditory scene analysis [16]. Indeed, Blinkies can potentially measure the spatial cepstrum which has been proposed as a robust and reliable feature [17]. Localization is another example where having sensors over a larger area allows localizing sources more precisely. Indeed, compact arrays are not able to gauge the distance to sources in the far field. Finally, there are a number of scenarios where joint use of conventional microphones and Blinkies has been already shown beneficial for speech enhancement [15] and separation [18], [19]. A number of specific use-cases are discussed in Section I-B.

The main thrust of this work is to explain in details the design and operation of Blinkies. We describe the hardware design of our prototype system, including trade-offs and rationals behind the choice of components, possible extensions of functionalities, as well as the choice of a camera. We analyze the whole signal path between sound acquired at a Blinky's microphone and signal recorded by the video camera. Accordingly, we devise a data encoding strategy to maximize the information received at the camera as explained. Because the brightness of each Blinky is attenuated differently by the channel, we propose a calibration strategy to restore the relative scales of all the signals. To validate the proposed system, we perform a live source localization experiment with a hundred-and-one Blinkies we built. We show that we are able to precisely localize a source in an environment with reverberation time over 1.5 s.

The remainder of this paper is organized as follows. Section II covers the system design, including hardware and software aspects. The signal path analysis is in Section III. Section IV explains the data encoding strategy. Calibration is discussed in Section V. The experimental setup and results are in Section VI. Section VII concludes this paper.

A. CHALLENGES

To make the proposed platform as useful as possible to the community, we included features beyond the strict needs of the current work. For example, while we advocate optical communication, the device itself is Wi-Fi enabled. It also has two microphones and four LEDs, while the strict minimum would be one of each. We hope that these provisions will extend the relevance of the device to a larger audience.

The encoding of the data to reduce distortion at the reception is an important topic. We propose a simple scheme described in Section IV that naturally allocates more bits to signal intensities more represented in the target audio signals. For many applications, it is crucial that the measurement from the sensors are corrected for the light attenuation occurring between sensor and camera. We show that this is possible and discuss different strategies in Section V.

Algorithmically speaking, the vast majority of array signal processing literature relies on phase information. In contrast, the proposed system has a sampling frequency in the tens of Hertz which renders most conventional algorithms unsuitable. To tackle this issue, we measure signal power for which a number of algorithms have been proposed over the years in the areas of low-rate and asynchronous systems [17], [20]–[23]. In addition, we previously proposed algorithms for scenarios mixing Blinkies and conventional microphones [15], [18]. Localization with 101 Blinkies is demonstrated in Section VI-B.

Finally, a number of challenges arise due to the nature of the optical communication channel. While left for future work, we enumerate them here for completeness. Obviously, due to human activity, numerous sources of interference exist in the visible light spectrum: daylight, house lights, etc. At the same time, Blinkies using visible light for operations can be distracting, or even obnoxious, to bystanders. A simple way around these problems is to use infrared LEDs in deployments. An extra challenge is that of sensor occlusion when an object stands between Blinky and camera.

While no work has been done in that direction yet, we believe that with a sufficiently large number of Blinkies, algorithms could be made insensitive to short occlusions of some of them.

B. APPLICATION SCENARIOS

While motivated by some of the same applications as conventional audio signal processing, the system proposed differs sufficiently that its applications are not completely obvious. We include here a non-exhaustive list of possible applications. We focus specifically on areas where sound has already been found to be effective.

1) SMART SPACES

Blinkies can be used to add perception of sound field to smart spaces. In the home, a camera-equipped voice assistant becomes aware of the distribution of sound sources in the room. In an office or conference setting, they can be distributed around locations where public speaking happens, allowing to enhance the audio of video recordings without extra equipment [15]. In particular, this could be used to improve the audio of videos taken with smartphones. Finally, museums could use them to provide interactive experiences to visitors in addition to straight security monitoring.

2) FACTORY MONITORING

Factories typically contain expensive machines and equipment that need to run with high availability. Monitoring is required to ensure the swift detection of malfunction or failure, and otherwise ensure production efficiency. Smart factories have been proposed to tackle these challenges and in particular internet-of-things (IoT) is seen as an enabling technology [24], [25]. While many IoT-based solutions rely on specialized sensor for monitoring, sound carries a lot of information regarding the state of operations and mechanical integrity of machines. It has been used in a variety of areas including gas and pipeline leak detection [26], [27], as well as structural integrity of equipment [28]. Combined to the frequent availability of CCTV cameras for site surveillance, this means that a number of smart factory tasks could be handled by only adding Blinkies at critical locations. They appear thus as an attractive, polyvalent, and cost-cutting solution.

3) TRAFFIC MONITORING

Traffic engineering is the discipline tasked with designing efficient roads and highways network systems [29]. Monitoring of said networks is necessary to identify bottlenecks and devise countermeasures. In addition to conventional induction loop vehicle counter [29], video [30] and audio [31] based systems have been proposed. A Blinky based system would benefit from the existing camera infrastructure while enabling vehicle detection in low light and challenging weather conditions. It would also be less computation and data-intensive than video based systems.

4) SECURITY

Besides obvious applications to gunshot [32] and car crash detection [33], monitoring elderly people living alone, for example for falls, is increasingly relevant in our aging society [34]. Due to privacy concerns, placing cameras or sending raw audio in homes is generally not possible. In contrast, sound power is sufficiently informative for efficient monitoring while avoiding privacy leaks.

While ubiquitous when it comes to surveillance, cameras are less useful in dark environment or in the presence of obstruction. Sound on the other hand travels around obstacles and sound-to-light sensors will work in dark conditions. The proposed sensor thus complements well existing security installations that might not be equipped with sound to provide a level of acoustic monitoring on top of visual.

5) DISASTER AREA MONITORING

During natural disasters, it is necessary to quickly deploy monitoring equipment over potentially large and remote areas. Such networks can be used for examples to locate survivors and guide rescue efforts. An alternative to wireless sensor networks (WSN) [35] could be to air drop robustified Blinkies to monitor sound in the disaster area. The signal acquisition can subsequently be done with a manned or unmanned aerial vehicle.

C. RELATED WORKS

In terms of hardware systems converting sound to light, there is a number of precedents. An early example is for the visualization of sound fields [36]. More recently, a system akin to Blinkies but based on analog technology was used to monitor frogs chorus [37]–[39]. Pablo Nava *et al.* proposed a microphone array using multiple LEDs per sensor and a high speed camera for light-based digital transmission of the audio signal [40]. It has also been proposed to use light-based signaling to synchronize arrays of asynchronous microphones [41].

Spatial sound power measurements, as collected by Blinkies, have already been proposed for a few applications. Several sound source localization algorithms exist in the context of wireless sensor networks [20], [21], [42]–[44] and speaker localization in meetings [45], [46]. A couple of speech enhancement techniques have been developed [22], [23]. Finally, the spatial cepstrum has been show to be effective for acoustic scene analysis [17].

Because Blinkies carry out some computations directly, they belong to the edge computing paradigm [47]. Neural processing has been shown to be viable in edge computing [48] and could potentially replace sound power for specific applications, for example acoustic event detection [49].

D. CONTRIBUTIONS

- The new open source hardware platform Blinky for large scale acoustic sensing. This contribution includes the circuit schematics, bill of materials, firmware, and the

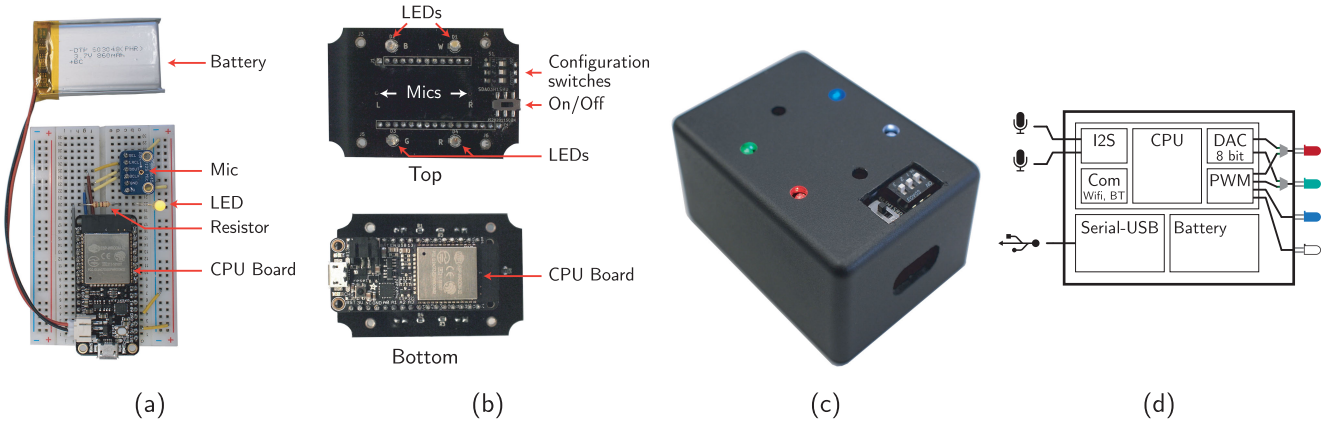

FIGURE 2. (a) A Blinky prototype built on a breadboard from off-the-shelf components. (b) Top and bottom view of the custom PCB for our Blinky design. (c) View of the Blinky with enclosure. (d) System diagram and wiring of the components to the CPU.

accompanying documentation in the companion material to this paper. We also share a graphical user interface (GUI) software for easy data acquisition of Blinky signals.
- We describe in details the channel between the sensor microphone and the signal collected by the video camera and propose a data encoding strategy accordingly.
- We propose and validate a calibration procedure to compensate for the channel attenuation.
- We demonstrate sound source localization with a hundred-and-one Blinkies in highly reverberent conditions

II. SYSTEM DESIGN

The acoustic sensing system we are designing is composed of three parts.

- The sensor, that we call Blinky, is equipped with a microphone and LED and converts the sound intensity to light intensity.
- A video camera simultaneously acquires the light emitted by multiple Blinkies in a video signal.
- A specialized video processing software extracts the individual sensor signals and records them to a file for later processing.

In the rest of this section we discuss in details the design choices made for each of these components.

A. SENSOR DESIGN

At a high level, a functional sound-to-light sensor requires a minimum of three components

- a microphone to capture sound,
- an LED to emit light,
- processing power.

In fact, recently, these three items do in fact correspond to commercially available hardware components that can be purchased and easily assembled into a working sensor (see Fig. 2 (a)). In addition to these fundamental requirements, we would like the sensor to be battery powered so that many devices can be used with minimum hassle. Finally, the sensor should be robust and easy to operate.

We made two specific choices to reduce the design complexity to a minimum. First, we decided to use a digital microphone requiring only minimal external components. Second, we built our prototype around a pre-assembled processor board including power, battery, and communication management. In fact, our prototype is so simple that it can be assembled on a breadboard with only four commercially available discrete components, as shown in Fig. 2 (a). In our final prototype, the microphones, LEDs, as well as a few switches used for easier operations, are assembled on a custom printed circuit board (PCB) which docks onto the pre-assembled processor board. The circuit is then housed in an enclosure with custom holes drilled for the microphones and LEDs to stick out. The PCB and final prototype are shown in Fig. 2 (b) and (c). We describe in more details the choice of the key components in the next few subsections.

1) PROCESSOR

We selected the ESP32 processor from ESPRESSIF SYSTEMS because it combines the right features with a very low cost and a wild popularity [50]. Its central processing unit (CPU) has two cores running at 240 MHz, 520 kB SRAM, 4 MB of flash memory, and a floating-point unit. These specifications are good enough to run fairly sophisticated processing in real time. It comes with a number of hardware peripherals, including two inter IC sound (I2S) modules allowing connecting audio peripherals easily, pulse-width modulation (PWM) to control the brightness of LEDs, and a dual channel 8 bit digital-to-analog converter. A diagram of the CPU and its peripherals is shown in Fig. 2 (d). The market price for a raw module is around USD 3 at the time of writing of this paper. In addition, a number of breakout boards designed around the ESP32 module and including USB-serial conversion, power regulation, and battery charging circuits are available for as low as around USD 7, e.g. WEMOS D32 [51]. For our prototype, we selected the HUZZAH32

board from the well-known open source hardware maker Adafruit [52].

Two other important criteria for choosing this platform were the availability of documentation and long term support. Due to its high popularity, in addition to the detailed official documentation, there is a profusion of tutorials, videos, and projects shared online. The processor is programmed in C++ via the official software development kit (SDK) or the popular Arduino IDE [53], and also supports Micropython [54] (a port of Python for embedded devices). Last but not least, ESPRESSIF system has committed support for the ESP32 until January 1st, 2028 [55].

2) MICROPHONES

In the last few years, MEMS microphones have become widely available and provide high quality sound input at low cost and tiny form factor. Moreover, whereas analog microphones would require an operational amplifier, some signal conditioning, and analog/digital conversion, modern digital MEMS come with everything included in a tiny package. In particular, a few models output the Inter-IC Sound (I2S) standard directly, such as the ICS-43432, from InveSense, that we selected. This is especially convenient as they can be directly wired to the ESP32 and provide pulse code modulation (PCM) audio format directly. Because one I2S bus is stereo, we decided to equip Blinkies with two microphones. This opens the door to simple array processing, such as DOA or null beamforming, for example.

3) LEDs

A major requirement for light-based communication is that the luminosity of the LED be not too dependent on the pose angle with respect to the camera. Surface mount technology (SMT) LEDs have a wide aperture, and high luminosity at low current. However, when mounted on a PCB inside an enclosure, even with a proper hole, their aperture is dramatically reduced. This could be mitigated by using Plexiglas light conductor at an increased assembly cost and design complexity. Instead, we opted for through-hole LEDs that can be elevated to stick out from the enclosure. While typical through-hole LEDs have an aperture of only 30 degrees, they can be made visible from nearly any angle by using diffuser caps. The aperture provided by different LED systems is illustrated in Fig. 3.

Another aspect to consider is the color of the LEDs. Conventional cameras typically use a layer of microfilters on top of the imaging sensor to acquire the red, green, and blue layers [56]. By using LEDs of various colors, it might be possible to take advantage of these filters to perform information multiplexing. With this vision in mind, we decided to equip each Blinky with four LEDs: white, red, green, and blue. As shown in Fig. 2 (d), all four LEDs are connected to the PWM. Two of the LEDs are in addition connected to the digital-analog converter (DAC) module which allows to run them at audio rates if necessary. Whether to use PWM or DAC can be selected in software.

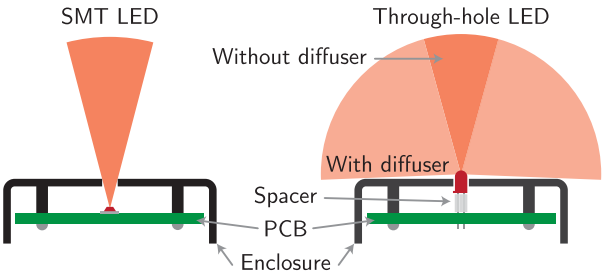

FIGURE 3. Illustration of surface mount versus through-hole LED visibility. When in an enclosure, SMT LEDs tend to sit too deep and thus have a poor aperture. Through-hole LEDs can stick out of the enclosure and use a diffuser cap to widen their aperture.

4) BATTERY

The device should be battery-operated with sufficient charge to last for several hours, or days, depending on the application. We selected rechargeable (Li-Poly) battery over regular alkaline dry cells. While the initial cost is higher, it amortizes over time. In addition, we do not need to carry large stocks of spare batteries for potentially hundreds of devices. Opting for USB rechargeable batteries allows the device to be charged with widely available smartphones chargers. For the prototype, we used a 400 mA h battery. Without enabling any of the power saving features of the processor, the device could be used for approximately half a day on a full charge. The battery life can be easily extended by enabling said features, or selecting a larger battery.

5) ENCLOSURE

We chose to house the Blinky in an enclosure to add robustness and visual appeal. It protects the circuit from physical damage and sensitive electrical components from direct contact. It also prevents excessive accumulation of dust on the components.

B. DESIGN EXTENSIONS

1) INFRARED LEDs

While visible light LEDs stand out for demonstration and make it easy to explain how the device works, they get irritating for users after some time. Replacing them by infrared LEDs would make the device essentially invisible while still being observable with many consumer cameras.

2) WIRELESS

Thanks to the Wi-Fi and Bluetooth capabilities of the processor, we can imagine different extensions and improvements of the device. We have already implemented over-the-air firmware update in the current device. Upon boot, it looks for a specific Wi-Fi network and server and checks if its own firmware is up-to-date. If a new firmware is available, it is installed and the device reboots. In addition, by changing the firmware, it is possible to turn the device into a wireless microphone that can then be re-used for a different style of distributed array processing. Finally, we can imagine using

Wi-Fi to transmit sound power in situations where light cannot work.

C. VIDEO ACQUISITION SYSTEM

The concurrent acquisition of the light signals emitted by the Blinkies is carried out by a video camera. It is a critical part of the system that needs to be carefully selected. The two most popular camera sensor types are charge-coupled device (CCD) and complementary metal-oxide-semiconductors (CMOS) sensors [57]. CCD has generally higher performance and is preferred for professional equipment, while CMOS is prevalent in consumer electronics due to its lower price. There are also two types of shutter mechanisms. Global shutter types read out all pixels simultaneously and so-called rolling shutters read rows sequentially [58]. For color imaging, a Bayer filter is placed on top of the sensor to split the light into red, green, and blue pixels [56]. While this could allow discriminating to some extent LEDs of different colors, it also reduces the amount of light acquired. A monochrome sensor should be favored for more precise measurements.

A small but important detail is whether the camera provides manual control of the acquisition parameters such as sensor sensitivity, exposure, etc. Many consumer-level video cameras dynamically adjust the sensitivity which might jeopardize measurements when done unexpectedly. On top of this, such cameras usually apply *gamma correction* to the recorded light energy, a lossy dynamic range compression operation described in Section III-B. Cameras providing the option to fix the sensor sensitivity should be preferred. In general, industrial cameras are more flexible and provide raw sensor data.

The frame rate of cameras vary from typically 30 frames-per-second (FPS) for consumer models to over 10000 FPS for specialized equipment [59]. Due to our focus on low communication overhead and price, we focus our attention on the low range between 30 and 60 FPS which is readily available in off-the-shelf video cameras. For applications requiring sampling close to audio rates, models in the 1000 FPS or above range can be selected at a higher cost.

Finally, in addition to garden-variety cameras, there exists a menagerie of specialized cameras available for more specialized deployment of Blinkies. Here are just a few examples. Infrared cameras can be used together with IR LEDs to make the Blinkies invisible. Panoramic surveillance cameras allow to monitor a large number of Blinkies in a tight indoor space. Stereo or multi-view cameras allow the localization of Blinkies using computer vision techniques [60] which can be useful for subsequent source localization in space.

We experimented with two different cameras. The Shodensha DN3V-130BU, a monochrome industrial camera with global shutter, and the Sony HDR-CX535, a color consumer camera with rolling shutter and no access to raw frames. As detailed in Section VI, the consumer camera could be used for simple applications, but for calibrated measurements of the light intensities, the industrial camera and its wider dynamic range was needed.

D. SOFTWARE

A straightforward approach to collecting data from Blinkies goes as follows. First, record a scene that contains Blinkies with a camera and save the video to a file. Second, find the pixel locations of the Blinkies in the video. Third, process the video to extract only the value of the pixel at the Blinky location, or a small patch around it.

This naive procedure has several drawbacks. Video files are in general very large, especially for raw format. Storing the whole file before discarding most of it is wasteful, or might be impractical. In addition, transfer between devices may be long or cumbersome. Another problem is finding the Blinkies in the video. Depending on the lighting conditions — the scene might be very dark, e.g. to allow sufficient dynamic range of the Blinkies — it might be difficult, if not impossible, to visually locate the devices. Finally, identifying the correct settings for the camera for optimal acquisition might require several iterations of this lengthy process.

FIGURE 4. A screen capture of the GUI developed for easy data acquisition with Blinkies. On the top left is the video stream. On the top right is the time signals of selected pixels. Just below, middle right, is the zoomed-in area. At the bottom are the controls and console.

We overcame these challenges by creating a GUI for data acquisition with Blinkies. A screen capture of the GUI is shown in Fig. 4. Based on `OpenCV` [61], the GUI allows to process both live camera streams or to operate on pre-recorded files. The live or recorded video is played in a large monitor window. A few buttons are available to show saturated pixels, or to apply a log transformation that makes objects in very dark videos visible. Pixels of the video can be selected by clicking on them. The last few seconds of the time signal of the selected pixel is then live streamed in a graph on the top right. The area around the clicked pixel is also displayed zoomed in on the right. By clicking in the zoom monitor, the choice of the pixel can be refined.

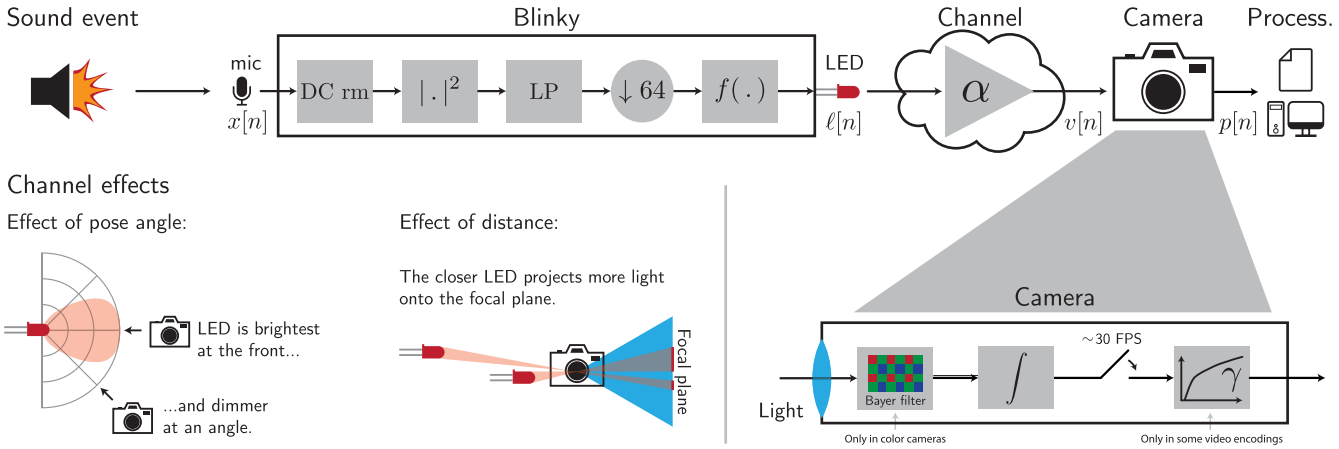

FIGURE 5. Overview of the Blinky acquisition system model. On top, the signal path from sound event to video file is detailed, including the processing in the Blinky. On the bottom left, the two factors impacting the light transmission between the Blinky and camera are illustrated. On the bottom right, the processing internal to the camera is shown.

Multiple pixels can be selected this way. Then, by pressing a button, it is possible to record the signals of only the selected pixels, or small patches around them. The signals recorded this way are saved to a file in the MessagePack format [62], which is both standard and efficient for binary data. The whole software is written in Python and thus easily runs on any operating system.

III. SYSTEM MODEL

The relationship between the sound incident onto the microphone and the corresponding pixel recorded by the video camera can be summarized by breaking down the system in three parts. In a Blinky, the sound is recorded and digitized as signal $x[n]$, with n being an arbitrary time index. This digital signal is processed and pre-conditioned to obtain a PWM duty cycle to drive the LED. Then, the light emitted by the LED is attenuated due to a combination of distance and incidence angle. Finally, the video camera sensor records the light and encode it in a video file format. This section focuses on the last two stages while the data encoding in the Blinky is covered in Section IV.

A. CHANNEL

The brightness of the LED is determined by the B-bit PWM duty cycle, $\ell[n] \in \{0, \dots, 2^B - 1\}$, set in the Blinky. The actual emitted light intensity is given by

$$I[n] = \frac{\ell[n]}{2^B - 1} I_{\max} \tag{1}$$

where $I_{\max}$ is the intensity of the LED driven continuously.

The two major factors affecting the light propagation between the LED and camera are the distance and pose angle between the two. An illustration is provided in Fig. 5. Let us first consider the distance. One can consider the LED to emit a fixed number of photons per time unit at a given brightness. The intensity recorded by the camera is then directly proportional to the number of photons hitting the detector. Naturally, when moving the detector further away from the LED, fewer photons hit the fixed surface of the detector and the recorded intensity is smaller.

The second factor is due to LEDs emitting photons unevenly, as mentioned in Section II-A.3. They are brightest when aligned with their radial symmetry axis, and gradually get dimmer when looked at an increasing angle. The angle at which the intensity drops by half is typically 10∼15°. Thus, when the camera is looking at a Blinky from an angle, it will record a lower intensity. In practice, both factors can be bundled into a single attenuation factor denoted by α.

Finally, ambient light reflected on the LED will add a positive bias β to the measured values. In Section V, we describe a procedure to calibrate α and β.

B. VIDEO PROCESSING

The light impinging on the lens, $v[n]$ in Fig. 5, is focused on the sensor. For color cameras, the sensor is further covered by a Bayer filter splitting the light into red, green, and blue (RGB) channels [56], $v_r[n]$, $v_g[n]$, and $v_b[n]$, respectively. The light energy is integrated over short intervals before being sampled at the frame rate (typically 30 or 60 FPS). This integration process is equivalent to a moving average in signal processing terms. Cameras for industrial applications usually provide the raw video frame without further processing. However, consumer cameras will in general apply a video compression scheme. Most video formats apply *gamma correction* to the recorded light energy. The exact operation depends on the format, but a popular function is $x^{1/\gamma}$ with $\gamma = 2.2$ [63]. Thus, the pixel value is $p[n] = (v[n])^{1/\gamma}$ for a monochrome camera. For a color camera, the RGB triplet is given by

$$\boldsymbol{p}[n] = [(v_r[n])^{1/\gamma}, (v_g[n])^{1/\gamma}, (v_b[n])^{1/\gamma}]. \tag{2}$$

It should be noted that the value of the pixels might be further affected by the video compression format [64].

However, we did not find these other effects significant in our experiments.

IV. DATA ENCODING

Between the audio input of the Blinky and the signal acquired by the camera, there are two lossy operations that we must take into account to avoid artefacts. The first is the mapping of the sound power range to the discrete PWM duty cycles. The second is the large difference of sampling rate between the audio and video at 16 kHz and 30 Hz, respectively.

The 24-bit audio signal provided by ICS-43432 microphone is converted to a floating-point value for convenience. First, a notch filter is used to remove any DC offset the signal might have. The output of this filter is used to compute the instantaneous power of the audio signal. The power envelope is obtained by low-pass filtering the instantaneous power of the output of the filter at a suitable rate for acquisition by the video camera. The low-pass filter is described in detail in the next section. The filtered signal is then decimated by 64 before applying an optional range compression function, $f(.)$ in Fig. 5. How to choose f is discussed in Section IV-B. The value thus obtained is quantized to the closest integer, giving the PWM duty cycle to drive the LED. This process is illustrated in Fig. 5.

A. LOW-PASS FILTERING

Within the Blinky, the instantaneous sound power is collected at 16 kHz. However, the camera sensor only sample lights at approximately 30 Hz and it is necessary to reduce the sample rate by computing the power envelope of the sound signal. A simple way to do that is by averaging power over short frames. The computation of the power envelope is akin to low-pass filtering and should ensure that the transmitted signal is not overly affected by aliasing.

In our implementation, the Blinky averages the power over 64 samples before updating the PWM duty cycle of the LED. This corresponds to an update frequency of 250 Hz of the LED. In addition, the light accumulation in the sensor of the camera also acts as an extra moving average of approximately $1/30$ s for a camera recording at 30 FPS. It was empirically determined that aliasing was not problematic in this configuration. An alternative to moving average could be infinite impulse response filters combining low-complexity with a sharper transition band [65].

B. RANGE COMPRESSION FUNCTION

Here we explore some coding strategies that can be applied to Blinky signaling. We would like to reduce the average error between the measured sound power and the signal received at the video camera. If we map the sound power linearly to the PWM range, all the small amplitude components of speech will be mapped to the lowest bits of the PWM range, resulting in a lot of inaccuracies. The larger elements which are relatively rare in speech would be assigned a disproportionate number of bits. Instead, what we would like to do is assign more bits to more frequent elements. What we propose is to first apply a non-linear transformation to make the speech power uniformly distributed. Then, by standard coding theory, uniform quantization can be applied to minimize the mean-squared error [67].

In practice, however, we do not know the optimal non-linear mapping, but we know that applying their cumulant distribution function (CDF) to samples from a given distribution will make them uniformly distributed. Because we also do not know the CDF of the sound power, we need to use a proxy to estimate it. As an example, we use the TIMIT corpus [66] to estimate the CDF of speech power. We break the speech samples into blocks of 4 ms, as in the Blinkies, and compute their variance. We remove all blocks with variance very close to zero as they correspond to silence. We then form the empirical CDF of the remaining and use a piecewise linear approximation to make it a continuous function. The histogram and CDF obtained are shown in Fig. 6. We see that by applying the inverse CDF, very small and very large values that are relatively infrequent will be mapped to a narrower range, as expected.

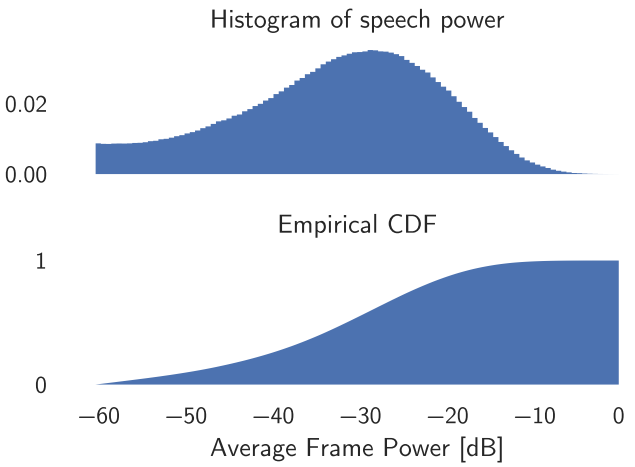

FIGURE 6. The empirical cumulative distribution function of the variance of short speech blocks computed on the TIMIT corpus [66].

While the presented methodology was used to obtain the compression function used in the experiments of Section VI, it might not be the optimal choice in general. Indeed, the speech samples in the TIMIT dataset are all recorded when the speaker is relatively close to the microphone, which is not realistic for Blinkies. In real environment, speech signals might be arbitrarily attenuated due to the distance between the source and the Blinky. A simple strategy to obtain a better CDF would be to place several Blinkies in a target environment and modify them to compute the microphone input power CDF over a long period. Then, the above procedure can be applied to this new dataset.

C. POTENTIAL FOR EDGE COMPUTING

In this paper, we have concentrated on using Blinkies for sound power measurements. However, they pack enough computing power to perform more complex processing.

A simple example is to add a denoising step so that only signals of interest are included in the sound power computation. This could be done by traditional means [68] or with a neural network [69].

V. CALIBRATION

As described in Section III-A, channel propagation and ambient lights add to each Blinky measurement a different attenuation α and offset β, respectively. For some applications, e.g. voice activity detection [15], this is not a problem. However, when correct relative sound power measurements are required, it is necessary to estimate the values of α and β. The calibration is also important if a non-linear range compression function, as described in Section IV-B, was used and needs to be inverted. An example application that requires this is the sound power separation by non-negative matrix factorization [19].

There are several ways to achieve the calibration, but all rely on having the Blinky transmit a known pilot signal. We consider the simplest case with an on/off signal whereas the LEDs is periodically turned off and then driven at a known duty cycle $\ell_{\text{ref}} \in \{0, \dots, 2^B - 1\}$. For example, this can be done in a calibration step at the beginning. Alternatively, Blinkies could periodically go into calibration mode and transmit the known signal for a short duration, e.g. every tens of minutes. Finally, a second auxiliary LED can be set to continuously transmit the pilot signal. We will now describe this last strategy in more details, as it is the most general.

The top of Fig. 5 summarizes the propagation model from sound power to pixel value. First, we need to estimate the luminosity of the LED from the pixel values recorded by the camera. To get a better estimate, we can sum the luminosity over several pixels illuminated by the target LED. We denote the set of these pixels by $\mathcal{A}$.

For monochrome, raw video frames, we can directly estimate the amount of light in the ith pixel by its value, i.e. $\hat{v}_i = p_i$, for $i \in \mathcal{A}$. If gamma correction was applied to the recording, it must be inverted, i.e. $\hat{v}_i = p_i^\gamma$, for all $i \in \mathcal{A}$. For color cameras, we further need to sum the light of all color channels, i.e. $\hat{v}_i = \hat{v}_i^r + \hat{v}_i^g + \hat{v}_i^b$, where $\hat{v}_i^r$, $\hat{v}_i^g$, and $\hat{v}_i^b$ are the value of the red, green, and blue channels, respectively, possibly with gamma correction inverted .[1] The final value is obtained by summing over $\mathcal{A}$,

$$\hat{v} = \sum_{i \in \mathcal{A}} \hat{v}_i. \tag{3}$$

Let ℓ_{sig} and ℓ_{ref} be the PWM duty cycles of the signal and calibration LEDs, respectively. We assume the two LEDs sufficiently close so that α and β are the same. Then, by (1), the corresponding light intensities impinging on the camera sensor are

$$v_{\text{sig}} = \alpha \frac{\ell_{\text{sig}}}{2^B - 1} I_{\max}^{(\text{sig})} + \beta, \tag{4}$$

[1] In image processing, weights are often applied to each color channel to match the sensitivity of human vision. We do not use such weights since we would like to measure the physically correct luminosity of the Blinky.

$$v_{\text{ref-lo}} = \beta, \tag{5}$$

$$v_{\text{ref-hi}} = \alpha \frac{\ell_{\text{ref}}}{2^B - 1} I_{\max}^{(\text{ref})} + \beta, \tag{6}$$

where v_{sig} is for the signal LED, and $v_{\text{ref-lo}}$ and $v_{\text{ref-hi}}$ are for the low and high levels of the calibration LED, respectively. Here, we assume that the signal and calibration LEDs might have different maximum intensities, $I_{\max}^{(\text{sig})}$ and $I_{\max}^{(\text{ref})}$, respectively, which can happen if they are of different physical construction. We assume these maximum values to be the same on all devices with the same construction. Our estimate of ℓ_{sig} is then

$$\hat{\ell}_{\text{sig}} = \frac{\hat{v}_{\text{sig}} - \hat{v}_{\text{ref-lo}}}{\hat{v}_{\text{ref-hi}} - \hat{v}_{\text{ref-lo}}} \ell_{\text{ref}} \approx \ell_{\text{sig}} \frac{I_{\max}^{(\text{sig})}}{I_{\max}^{(\text{ref})}}. \tag{7}$$

In case $I_{\max}^{(\text{sig})} = I_{\max}^{(\text{ref})}$, we recover ℓ_{sig} exactly. In the alternative, a global scaling occurs. When undesirable, it is possible to measure $I_{\max}^{(\text{sig})}$ and $I_{\max}^{(\text{ref})}$ in advance and simply scale the result by their ratio.

VI. EXPERIMENTS

As a proof-of-concept for the Blinky system presented in this paper, we show the results of two experiments. The first one demonstrates recovery of correct audio levels at different locations. In the second, we perform localization of a moving sound source with a hundred-and-one actual Blinkies.

A. SOUND POWER ESTIMATION

Our aim in this experiment is to validate the calibration procedure described in Section V. To do so, we playback sound and simultaneously record the sound power and waveform with Blinkies and microphones placed within a few centimeters, respectively. We then measure the discrepancy between the sound power measured by Blinkies and microphones.

1) EXPERIMENTAL SETUP

We used the Shodensha DN3V-130BU industrial camera placed in an office at 6.6 m and 7.6 m from two Blinkies, respectively. During a calibration phase, the Blinkies were set to cycle the LED between full on and off with a period of 3 s and half duty cycle, as described in Section V. The software described in Section II-D was used during this phase to locate the Blinkies within the video frame and set the aperture of the camera so that the LEDs did not saturate the camera sensor. The two Blinkies were located at pixels (861, 174) and (1003, 576), respectively, in the 1280×1024 video frame. We setup the system to record patches of 3×3 pixels around the locations of the sensors that were then averaged to estimate the light intensity. We recorded 30 s of signal in this mode. Then, we ordered all the recorded values and separated the bottom and top 50 %. Finally, we estimated $\hat{v}_{\text{ref-lo}}$ as the minimum of the lower half, and $\hat{v}_{\text{ref-hi}}$ as the median of the top half.

In the second part of the experiment, we placed a sound source at 2.17 m and 0.84 m from the Blinkies, respectively. The Blinkies were then switched from calibration to sound

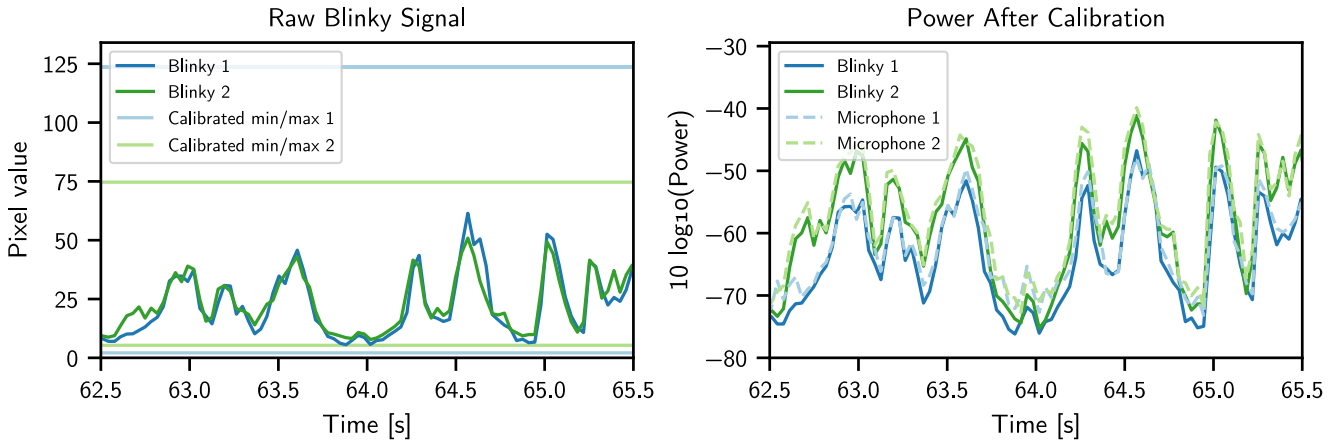

FIGURE 7. Comparison of the sound power recorded by two Blinkies and two microphones placed within a few centimeters. On the left are the uncalibrated Blinky signals together with the minimum and maximum sensor values obtained during calibration. On the right, we overlap the sound power measured by Blinkies and microphones. We observe that they match tightly.

level monitoring where the intensity of the LED is proportional to the sound level. In this mode, the following processing was applied in the Blinkies to compute the duty cycle of the LED from the microphone input. The audio input was normalized to be between -1 and 1. The sound power was then computed as described in Section IV and transformed to decibels with the reference being the maximum power, i.e. 1. The power was clipped between -80 dB and -10 dB as we determined empirically that most sounds in our environment fell in this range. The dynamic range compression function used was the empirical CDF from Fig. 6, but with the x-axis stretched to cover $[-80, -10]$. This choice is arbitrary and could easily be changed to a more appropriate function according to the target application.

The sound source played speech from the JNAS database [70] for approximately 2 min. During this time, the signal from the Blinkies was recorded with the same setup as the calibration. Simultaneously, we recorded the same sound signal using conventional microphones placed within a few centimeters of the Blinkies. We thus expect that the sound power recorded by the microphones should be very close to that recorded by the Blinkies and will use this as a reference signal. After recording, we applied (7) with $\hat{v}_{\text{ref-lo}}$ and $\hat{v}_{\text{ref-hi}}$ obtained during calibration to recover the sound level $P_{\text{blinky}}[n]$ from the Blinky signal.

Because the Blinkies, cameras, and microphones are all asynchronous, we needed to match the sampling frequency and time offset of the microphone signals to the camera's before comparing the signals. Then, the sound power measured by the microphone $P_{\text{microphone}}[n]$ was computed at the rate of the Blinky signal. We measure the accuracy of the calibration in terms of average Blinky-to-microphone power ratio (ABMPR) computed in the following way,

$$\text{ABMPR} = \frac{1}{T}\sum_{n=1}^{T}\left|10\log_{10}\frac{P_{\text{blinky}}[n]}{P_{\text{microphone}}[n]}\right|, \qquad (8)$$

where n and T are the index and number of samples, respectively. We further averaged over the two Blinky locations. With perfect calibration, we expect $P_{\text{blinky}}[n] = P_{\text{microphone}}[n]$ for all n which would result in ABMPR $= 0$.

2) RESULT

We find that ABMPR $=2.85$ dB. This means that the two power signals are within a factor two in average, which is sufficient for many applications based on sound power. Note that the microphone signal used as a reference here is not the true signal form the Blinky microphone and that some discrepancy is thus expected. Fig. 7 shows a 3 s extract from the signals before and after calibration. We observe that the calibration is excellent for larger power, but somewhat off for small amplitudes. While a more careful analysis would be needed to determine the cause, we conjecture that a more careful design of the range compression function could alleviate such problems. We can also see that the calibration faithfully restores the relative amplitude of sound at the different sensors that was lost due to the channel attenuation. In addition, the non-linear mapping is successfully inverted after propagation and acquisition. This demonstrates the proposed system is able to accurately measure sound power over space.

B. SOURCE LOCALIZATION WITH 101 BLINKIES

We will now demonstrate the use of Blinkies in a practical experiment of sound source localization. For this purpose, we deployed 101 Blinkies in a highly reverberant environment. Unlike our previous work [15], the localization is not done in space, but within the video frame captured by the camera. The goal is to recover the pixel coordinate of the sound source. We compare two methods of localization neither of which require calibration of the Blinkies. The first one uses the brightest Blinkies to estimate the location. The second is data-driven and uses a neural network trained on a subset of the collected data.

1) NOTATION AND PROBLEM STATEMENT

We assume M Blinkies with pixel coordinates $\boldsymbol{r}_m \in \mathbb{R}^2$, $m = 1, \ldots, M$. The Blinky signals ℓ_m can be recovered by extracting the pixel values around their locations in the video frame as in (3).

Assuming a sound source is located at $\boldsymbol{s}$ (in pixel space), the sound power measured at the m-th Blinky will be approximately inversely proportional to their distance (in real space). As the source moves in the scene, the brightness of the Blinkies will follow it. Thus, by concatenating the signals from all M Blinkies, we obtain a fingerprint of the sound source location.

2) BASELINE ALGORITHM

In the presence of a single sound source, the Blinky closest is usually also the brightest. This observation is the basis for our baseline localization algorithm. Namely, the estimated source location is the weighted average of that of the K brightest Blinkies

$$\hat{\boldsymbol{s}} = \sum_{k=1}^{K} \frac{\ell_{(k)}}{Z} \boldsymbol{r}_{(k)} \tag{9}$$

where $\ell_{(1)} \geq \ldots \geq \ell_{(K)}$ and $\{\boldsymbol{r}_{(k)}\}_{k=1}^{K}$ are the ordered signals and locations, respectively, of the Blinkies, and $Z = \sum_{k=1}^{K} \ell_{(k)}$ is a normalization factor. When $K = 1$, the algorithm assigns the location of the brightest Blinky to the source. Note that the weighting scheme is completely heuristic since we do not know the exact relationship between distance to a Blinky and the intensity of its signal.

3) LEARNING-BASED LOCALIZATION

A different approach is to learn the mapping between sound source location and Blinky intensities directly using a dataset set aside for this purpose. We create a neural network with M inputs and 2 outputs. The network consists of three stages of residual networks [71] sandwiched between input and output fully connected layers. The input layer is followed by 10% dropout for regularization [72]. Rectified linear units (ReLU) are used for activations. The loss function is the mean squared error. The network structure is shown in Fig. 8.

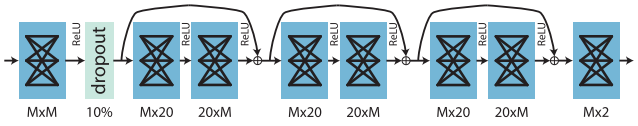

FIGURE 8. The neural network architecture used in the localization experiment. The inputs are the $M = 101$ sensor values and the outputs the x, y coordinates of the source in the video frame.

4) EXPERIMENTAL SETUP

The $M = 101$ Blinkies are deployed in a 34.3 m-by-29.3 m gymnastic hall with ceiling height between 8.7 m and 12.9 m, and reverberation time of over 1.5 s. The Blinkies were arranged in an approximate grid. A Sony HDR-CX535 camera was placed in the corner of the hall at a height of approximately 4 m from the ground. The video format was H264 with resolution 1440×1090 pixels at 30 frames per second. The sound source, a Bluetooth speaker, was mounted onto a radio-controlled (RC) model car. A red light bulb was added onto the car to provide its groundtruth location in the video. The car was then piloted on three different paths between the Blinkies, once parallel to each side of the grid, and once in diagonal. The setup is illustrated in Fig. 9. The experiment was repeated once with white noise and once with speech. In addition, five short segments with a static human speaker at distinct locations were recorded.

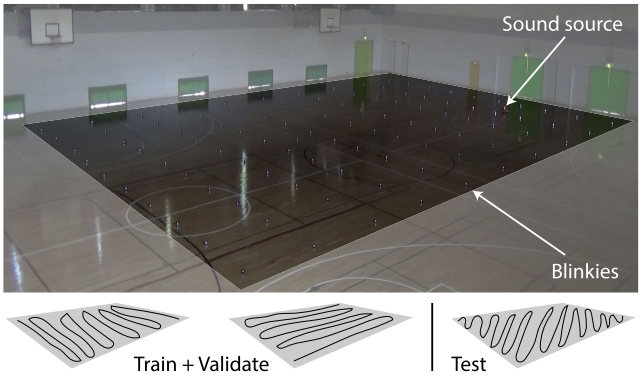

FIGURE 9. Top: a frame of the video (truncated to 1440 × 670) with source location and Blinkies highlighted. Bottom: diagrams of the paths of the RC car and their assignment to training/testing.

The locations of the Blinkies in the image were manually recorded and the pixels intensities in a 3×3 square around were averaged after inverse gamma correction with $\gamma = 2.8$. The data was split between parallel paths for training and validation (20739 and 2304 examples, respectively), and diagonal paths and human speaker for test (12727 and 905 examples, respectively). The neural network was trained with Chainer [73] and the Adam optimizer [74] for 3000 epochs and with a mini-batch size of 200.

TABLE 1. The 50-th and 90-th percentile of localization error (in pixels) of baseline (BL) and neural network (NN).

RC Car	**BL** $K=1$	$K=2$	$K=3$	**NN**
p_{50}	39.6	37.5	71.1	**29.6**
p_{90}	352.2	307.6	308.1	**99.2**
Human	**BL** $K=1$	$K=2$	$K=3$	**NN**
p_{50}	69.0	28.7	37.5	**22.9**
p_{90}	395.3	359.6	342.8	**130.1**

5) RESULTS

The results of the evaluation on the test sets (RC car and human speaker) is given in Table 1. The baseline method works best with $K = 2$ and achieves median error of 37.5 and 28.7 pixels, respectively. The neural network

achieves the best performance with 29.6 and 22.9 pixels error, respectively. In addition, the distribution of error is much more compact for the neural network. The 90-th percentile of its error distribution is 99.2 and 130.1, while it is always over 300 for the baseline method. Thus, we confirmed that sound source localization with Blinkies is effective, even in reverberant environments.

VII. CONCLUSION

We presented a new open source hardware platform for large-scale acoustic sensing that we call a Blinky. The sensor converts sound power to a luminous signal emitted by an LED. The signals from numerous Blinkies can be recorded simultaneously by a video camera. We described the hardware in details and also offer schematics, software, and documentation to reproduce the platform in the supplementary material. We discussed a number of possible applications of Blinkies along with their merits and limitations. We presented a model for the signal path and a method for the calibration of the sensors, which was experimentally validated. In an experiment with a hundred-and-one Blinkies, we demonstrated sound source localization in highly reverberent conditions. Due to their versatility and extensibility, we believe Blinkies can be of interest to a large community.

In future work, we would like to focus on applications of Blinkies, in particular to acoustic event detection. There are also a number of practical challenges, such as sensor occlusion, to which we would like to find algorithmic solutions. Beyond sound power, we are interested in applying sophisticated processing directly in the Blinky. We believe this could boost performance significantly, especially in challenging environments.

ACKNOWLEDGMENT

The authors thank Daiki Horiike for his precious help during the experiments. The hardware design files, schematics, firmware, and software for the Blinky are available at https://github.com/onolab-tmu/blinky. The code to reproduce the experiments in this article is shared at https://github.com/onolab-tmu/code_2020IEEEAccess_blinky. The data to reproduce the localization experiment is available at https://zenodo.org/record/3635217.

REFERENCES

[1] J. Schnupp, I. Nelken, and A. King, *Auditory Neuroscience: Making Sense Sound*. Cambridge, MA, USA: MIT Press, 2011.
[2] M. Brandstein and D. Ward, *Microphone Arrays* (Signal Processing Techniques and Applications). Berlin, Germany: Springer-Verlag, Dec. 2010.
[3] H. L. Van Trees, *Optimum Array Processing*. New York, NY, USA: Wiley, 2002.
[4] I. Dokmanic, R. Scheibler, and M. Vetterli, "Raking the cocktail party," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 5, pp. 825–836, Aug. 2015.
[5] S. Makino, Ed., *Audio Source Separation*. Cham, Switzerland: Springer, 2018.
[6] H. Krim and M. Viberg, "Two decades of array signal processing research: The parametric approach," *IEEE Signal Process. Mag.*, vol. 13, no. 4, pp. 67–94, Jul. 1996.
[7] E. Weinstein, "Optimal source localization and tracking from passive array measurements," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 30, no. 1, pp. 69–76, Feb. 1982.
[8] I. Dokmanic, R. Parhizkar, A. Walther, Y. M. Lu, and M. Vetterli, "Acoustic echoes reveal room shape," *Proc. Nat. Acad. Sci. USA*, vol. 110, no. 30, pp. 12186–12191, Jul. 2013.
[9] E. A. P. Habets and J. Benesty, "A two-stage beamforming approach for noise reduction and dereverberation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 21, no. 5, pp. 945–958, May 2013.
[10] T. Virtanen, M. D. Plumbley, and D. Ellis, *Computational Analysis of Sound Scenes and Events*. Cham, Switzerland: Springer, 2018.
[11] R. Scheibler, J. Azcarreta, R. Beuchat, and C. Ferry, "Pyramic: Full stack open microphone array architecture and dataset," in *Proc. 16th Int. Workshop Acoustic Signal Enhancement (IWAENC)*, Sep. 2018, pp. 226–230.
[12] E. Weinstein, K. Steele, A. Agarwal, and J. Glass, "LOUD: A 1020-node microphone array and acoustic beamformer," in *Proc. ICSV*, Cairns, Australia, Jul. 2007, pp. 1–5.
[13] F. Perrodin, J. Nikolic, J. Busset, and R. Siegwart, "Design and calibration of large microphone arrays for robotic applications," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2012, pp. 4596–4601.
[14] S. Miyabe, N. Ono, and S. Makino, "Blind compensation of interchannel sampling frequency mismatch for ad hoc microphone array based on maximum likelihood estimation," *Signal Process.*, vol. 107, pp. 185–196, Feb. 2015.
[15] R. Scheibler, D. Horiike, and N. Ono, "Blinkies: Sound-to-light conversion sensors and their application to speech enhancement and sound source localization," in *Proc. Asia–Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA ASC)*, Nov. 2018, pp. 1899–1904.
[16] G. J. Brown and M. Cooke, "Computational auditory scene analysis," *Comput. Speech Language*, vol. 8, no. 4, pp. 297–336, Oct. 1994.
[17] K. Imoto and N. Ono, "Spatial cepstrum as a spatial feature using a distributed microphone array for acoustic scene analysis," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 6, pp. 1335–1343, Jun. 2017.
[18] R. Scheibler and N. Ono, "Multi-modal blind source separation with microphones and blinkies," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 366–370.
[19] D. Horiike, R. Scheibler, Y. Wakabayashi, and N. Ono, "Blink-former: Light-aided beamforming for multiple targets enhancement," in *Proc. IEEE 21st Int. Workshop Multimedia Signal Process. (MMSP)*, Sep. 2019, pp. 1–6.
[20] X. Sheng and Y.-H. Hu, "Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 53, no. 1, pp. 44–53, Jan. 2005.
[21] D. Blatt and A. O. Hero, "Energy-based sensor network source localization via projection onto convex sets," *IEEE Trans. Signal Process.*, vol. 54, no. 9, pp. 3614–3619, Sep. 2006.
[22] H. Chiba, N. Ono, S. Miyabe, Y. Takahashi, T. Yamada, and S. Makino, "Amplitude-based speech enhancement with nonnegative matrix factorization for asynchronous distributed recording," in *Proc. 14th Int. Workshop Acoustic Signal Enhancement (IWAENC)*, Sep. 2014, pp. 203–207.
[23] Y. Matsui, S. Makino, N. Ono, and T. Yamada, "Multiple far noise suppression in a real environment using transfer-function-gain NMF," in *Proc. 25th Eur. Signal Process. Conf. (EUSIPCO)*, Aug. 2017, pp. 2314–2318.
[24] B. Chen, J. Wan, L. Shu, P. Li, M. Mukherjee, and B. Yin, "Smart factory of industry 4.0: Key technologies, application case, and challenges," *IEEE Access*, vol. 6, pp. 6505–6519, 2018.
[25] P. K. Illa and N. Padhi, "Practical guide to smart factory transition using IoT, big data and edge analytics," *IEEE Access*, vol. 6, pp. 55162–55170, 2018.
[26] P.-S. Murvay and I. Silea, "A survey on gas leak detection and localization techniques," *J. Loss Prevention Process Industries*, vol. 25, no. 6, pp. 966–973, Nov. 2012.
[27] S. W. Oh, D.-B. Yoon, G. J. Kim, J.-H. Bae, and H. S. Kim, "Acoustic data condensation to enhance pipeline leak detection," *Nucl. Eng. Design*, vol. 327, pp. 198–211, Feb. 2018.
[28] K. Ono, "Structural integrity evaluation by means of acoustic emission," in *Acoustic Emission and Critical Phenomena*. Didcot, U.K.: Taylor Francis, Feb. 2010, pp. 13–27.
[29] R. P. Roess, E. S. Prassas, and W. R. McShane, *Traffic Engineering*. Upper Saddle River, NJ, USA: Prentice-Hall, 2018.
[30] Z. Chen, T. Ellis, and S. A. Velastin, "Vision-based traffic surveys in urban environments," *J. Electron. Imag.*, vol. 25, no. 5, Apr. 2016, Art. no. 051206.
[31] B. Barbagli, L. Bencini, I. Magrini, G. Manes, and A. Manes, "A real-time traffic monitoring based on wireless sensor network technologies," in *Proc. 7th Int. Wireless Commun. Mobile Comput. Conf.*, Istanbul, TU, USA, Jul. 2011, pp. 820–825.

[32] J. Aguilar, "Gunshot detection systems in civilian law enforcement," *J. Audio Eng. Soc.*, vol. 63, no. 4, pp. 280–291, Apr. 2015.
[33] P. Foggia, N. Petkov, A. Saggese, N. Strisciuglio, and M. Vento, "Audio surveillance of roads: A system for detecting anomalous sounds," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 1, pp. 279–288, Jan. 2016.
[34] J. Wang, Z. Zhang, B. Li, S. Lee, and R. S. Sherratt, "An enhanced fall detection system for elderly person monitoring using consumer home networks," *IEEE Trans. Consum. Electron.*, vol. 60, no. 1, pp. 23–29, Feb. 2014.
[35] D. Chen, Z. Liu, L. Wang, M. Dou, J. Chen, and H. Li, "Natural disaster monitoring with wireless sensor networks: A case study of data-intensive applications upon low-cost scalable systems," *Mobile Netw. Appl.*, vol. 18, no. 5, pp. 651–663, Oct. 2013.
[36] W. E. Kock, *Seeing Sound*. Hoboken, NJ, USA: Wiley, 1971.
[37] I. Aihara, T. Mizumoto, T. Otsuka, H. Awano, K. Nagira, H. G. Okuno, and K. Aihara, "Spatio-temporal dynamics in collective frog choruses examined by mathematical modeling and field observations," *Sci. Rep.*, vol. 4, no. 1, May 2015, Art. no. 011918.
[38] T. Mizumoto, I. Aihara, T. Otsuka, H. Awano, and H. G. Okuno, "Swarm of sound-to-light conversion devices to monitor acoustic communication among small nocturnal animals," *J. Robot. Mechatronics*, vol. 29, no. 1, pp. 255–267, Feb. 2017.
[39] T. Mizumoto, I. Aihara, T. Otsuka, R. Takeda, K. Aihara, and H. G. Okuno, "Sound imaging of nocturnal animal calls in their natural habitat," *J. Comparative Physiol.*, vol. 197, no. 9, pp. 915–921, Sep. 2011.
[40] G. Pablo Nava, H. Duy Nguyen, Y. Kamamoto, T. G. Sato, Y. Shiraki, N. Harada, and T. Moriya, "A high-speed camera-based approach to massive sound sensing with optical wireless acoustic sensors," *IEEE Trans. Comput. Imag.*, vol. 1, no. 2, pp. 126–139, Jun. 2015.
[41] T. Akiyama, M. Sugimoto, and H. Hashizume, "Light-synchronized acoustic ToA measurement system for mobile smart nodes," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Oct. 2014, pp. 749–752.
[42] C. Meesookho, U. Mitra, and S. Narayanan, "On energy-based acoustic source localization for sensor networks," *IEEE Trans. Signal Process.*, vol. 56, no. 1, pp. 365–377, Jan. 2008.
[43] Y. Zhang, D. Xue, C. Wu, and W. Meng, "A new algorithm for multiple-source localization based on acoustic energy in wieless sensor networks," in *Proc. Int. Conf. Ind. Mechatronics Autom.*, May 2009, pp. 360–363.
[44] D. Ampeliotis and K. Berberidis, "Low complexity multiple acoustic source localization in sensor networks based on energy measurements," *Signal Process.*, vol. 90, no. 4, pp. 1300–1312, Apr. 2010.
[45] Z. Liu, Z. Zhang, L.-W. He, and P. Chou, "Energy-based sound source localization and gain normalization for ad hoc microphone arrays," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Honolulu, HI, USA, Apr. 2007, pp. 761–764.
[46] M. Chen, Z. Liu, L.-W. He, P. Chou, and Z. Zhang, "Energy-based position estimation of microphones and speakers for ad hoc microphone arrays," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, Oct. 2007, pp. 22–25.
[47] E. Baccarelli, P. G. V. Naranjo, M. Scarpiniti, M. Shojafar, and J. H. Abawajy, "Fog of everything: Energy-efficient networked computing architectures, research challenges, and a case study," *IEEE Access*, vol. 5, pp. 9882–9910, 2017.
[48] P. S. Chandakkar, Y. Li, P. L. K. Ding, and B. Li, "Strategies for re-training a pruned neural network in an edge computing paradigm," in *Proc. IEEE Int. Conf. Edge Comput. (EDGE)*, Jun. 2017, pp. 244–247.
[49] Y. Ngoko and C. Cerin, "An edge computing platform for the detection of acoustic events," in *Proc. IEEE Int. Conf. Edge Comput. (EDGE)*, Jun. 2017, pp. 240–243.
[50] (2019). *ESP32 Datasheet*. Accessed: May 31, 2019. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32_datas%heet_en.pdf
[51] WEMOS. *D32*. Accessed: May 31, 2019. [Online]. Available: https://wiki.wemos.cc/products:d32:d32
[52] Adafruit. *Adafruit HUZZAH32–ESP32 Feather Board*. Accessed: May 31, 2019. [Online]. Available: https://www.adafruit.com/product/3405
[53] *Arduino*. Accessed: Jun. 5, 2019. [Online]. Available: https://www.arduino.cc
[54] *Micropython*. Accessed: Jun. 5, 2019. [Online]. Available: https://micropython.org
[55] E. Systems. *Longevity Commitment*. Accessed: May 31, 2019. [Online]. Available: https://www.espressif.com/en/products/longevity-commitment
[56] B. E. Bayer, "Color imaging array," U.S. Patent 3 971 065 A, Jul. 20, 1976.
[57] E. R. Fossum and D. B. Hondongwa, "A review of the pinned photodiode for CCD and CMOS image sensors," *IEEE J. Electron Devices Soc.*, vol. 2, no. 3, pp. 33–43, May 2014.
[58] J. Nakamura, *Imagesensors and Signal Processing for Digital Still Cameras*. Bosa Roca, USA: Taylor & Francis, 2005.
[59] I. Ishii, T. Tatebe, Q. Gu, Y. Moriue, T. Takaki, and K. Tajima, "2000 fps real-time vision system with high-frame-rate video recording," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2010, pp. 1536–1541.
[60] R. Hartley and A. Zisserman, *Multiple view geometry Computer Vision*, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 2003.
[61] G. Bradski, "The open CV library," *Dr. Dobb's J. Softw. Tools*, 2000. [Online]. Available: https://www.drdobbs.com/open-source/the-opencv-library/184404319
[62] *Messagepack*. Accessed: Jan. 29, 2020. [Online]. Available: https://msgpack.org
[63] *Multimedia Systems and Equipment—Colour Measurement and Management—Part 2-1: Colour Management—Default Rgb Colour Space—Srgb*, International Electrotechnical Commission, Geneva, CH, USA, Oct. 1999.
[64] *ITU-T H.264 (V12): Advanced Video Coding for Generic Audiovisual Services*, International Telecommunication Union, Geneva, CH, USA, Apr. 2017.
[65] P. Prandoni and M. Vetterli, *Signal Processing for Communication*, 1st ed. Lausanne, Switzerland: EPFL Press, 2008.
[66] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, and D. S. Pallett, "DARPA TIMIT acoustic-phonetic continous speech corpus CD-ROM. NIST speech disc 1-1.1," Nat. Inst. Standards Technol., Gaithersburg, MD, USA, Tech. Rep. 4930, Tech. Rep., Feb. 1993.
[67] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Hoboken, NJ, USA: Wiley, Jul. 2006.
[68] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 33, no. 2, pp. 443–445, Apr. 1985.
[69] S. R. Park and J. W. Lee, "A fully convolutional neural network for speech enhancement," in *Proc. Interspeech*, Aug. 2017, pp. 1993–1997.
[70] K. Itou, M. Yamamoto, K. Takeda, T. Takezawa, T. Matsuoka, T. Kobayashi, K. Shikano, and S. Itahashi, "JNAS: Japanese speech corpus for large vocabulary continuous speech recognition research," *J. Acoust. Soc. Jpn.*, vol. 20, no. 3, pp. 199–206, May 1999.
[71] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[72] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, Jun. 2014.
[73] S. Tokui, K. Oono, S. Hido, and J. Clayton, "Chainer: A next-generation open source framework for deep learning," in *Proc. 29th Workshop Mach. Learn. Syst. Neural Inf. Process. Syst. (NIPS)*, vol. 5, Dec. 2015, pp. 1–6.
[74] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980

ROBIN SCHEIBLER (Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees from the École Polytechnique Fédérale de Lausanne (EPFL), Switzerland. He worked at the Research Labs, NEC Corporation, Kawasaki, Japan, and IBM Research, Zürich, Switzerland. He is a specially appointed Associate Professor with Tokyo Metropolitan University, Tokyo, Japan. In March 2020, he became a Researcher at LINE Corporation, Tokyo. His research interests are in efficient algorithms for signal processing and array signal processing more particularly. He also likes to build large microphone arrays. He is the Lead Developer of pyroomacoustics, an open source library for room acoustics simulation and array signal processing.

NOBUTAKA ONO (Senior Member, IEEE) received the B.E., M.S., and Ph.D. degrees in mathematical engineering and information physics from The University of Tokyo, Japan, in 1996, 1998, 2001, respectively. He joined the Graduate School of Information Science and Technology, The University of Tokyo, in April 2001, as a Research Associate, where he became a Lecturer in April 2005. He moved to the National Institute of Informatics, Japan, as an Associate Professor, in April 2011, where he became a Professor, in September 2017. He moved to Tokyo Metropolitan University, in October 2017. His research interests include acoustic signal processing, specifically, microphone array processing, source localization and separation, machine learning, and optimization algorithms for them. He is the author or coauthor of more than 240 articles in international journal articles and peer-reviewed conference proceedings. He has been a member of the IEEE Audio and Acoustic Signal Processing (AASP) Technical Committee, since 2014 and a Senior Member of the IEEE Signal Processing Society and a member of the Acoustical Society of Japan (ASJ), the Institute of Electronics, Information and Communications Engineers (IEICE), the Information Processing Society of Japan (IPSJ), and the Society of Instrument and Control Engineers (SICE) in Japan. He received the Sato Paper Award and the Awaya Award from ASJ, in 2000 and 2007, the Igarashi Award at the Sensor Symposium on Sensors, Micromachines, and Applied Systems from IEEJ, in 2004, the Best Paper Award from IEEE ISIE, in 2008, the Measurement Division Best Paper Award from SICE, in 2013, the Best Paper Award from IEEE IS3C, in 2014, the Excellent Paper Award from IIHMSP, in 2014, the Unsupervised Learning ICA Pioneer Award from SPIE.DSS, in 2015, the Sato Paper Award from ASJ, two TAF Telecom System Technology Awards, in 2018, and the Best Paper Award from APSIPA, in 2018. He was the Tutorial Speaker at ISMIR 2010 and ICASSP 2018, the special Session Chair in EUSIPCO, in 2013, 2015, 2017, 2018, and 2019, and the Chair of Signal Separation Evaluation Campaign (SiSEC) Evaluation Committee, in 2013 and 2015. He was an Associate Editor of the IEEE TRANSACTIONS ON AUDIO, SPEECH AND LANGUAGE PROCESSING, from 2012 to 2015.