Indian Institute of Technology Bombay

Real-Time Gunshot Localization Using Acoustics

EE 451: Supervised Research Exposition

Aditya Rajeev Harakare
Roll Number: 18D100001

Project Guide:
Prof. Rajbabu Velmurugan

Department of Electrical Engineering

Contents

1 Abstract

In the military and security forces, many times the location of the origin of the gunfire is required for making further tactical decisions. The security forces face difficulties in asserting this location especially in obstacle-dense and reverberate environments. It might not be possible for a person to stay alert all the time and locate any gunshot. However, in case of a gunfire, the source location is required immediately for prompt and effective counter-measures.
The source location in 2-D can be characterised by 2 parameters:

- Range
- Bearing Angle

Out of these 2 parameters, finding the bearing angle accurately is more important than the range for the given use case. Different algorithms for gunshot localization were surveyed in the project. The project is an attempt to design a simulator to simulate the gunshot events and the microphone array. A python implementation of localization using wavefront curvature method is described.
Github Link for the Project: `https://github.com/adityaharakare1/SREGunshotLoc`

2 Objectives of this Project

Following were the objectives of this project:

1. Understand the problem statement and use cases of gunshot localization.
2. Study the acoustic signatures of a gunshot event.
3. Survey the different algorithms used for gunshot localization using the acoustic signatures.
4. Propose hardware for implementing the solution.
5. Design a simulator in MATLAB for simulating the muzzle blast from the gunshot and the planar 4-microphone array.
6. Develop a Python code for gunshot localization using Wavefront Curvature Method and test it on RPi using the data from the simulator.
7. Study the effect of Noise and reflection artifacts on the location estimate.

3 Introduction

In a gunfire event, follwing signatures are left:

1. Sound of charge explosion in gun muzzle
2. Light of the charge explosion
3. Sound of the bullet travelling, which is only perceptible if the bullet's speed is supersonic
4. 'Sight' of the bullet travelling
5. Sound of the bullet hitting the target, and

6. Smell of the gunfire

Evidently, only the first and third signatures are useful for reliable localization of the shooter. We see that both of these are acoustic in nature.
The objective of this project was thus to develop a system that can robustly detect and localize gunshots based on their acoustic signature
Acoustic targeting devices can be classified based on their mobility:

- Static Devices
- Vehicle Mounted
- Helmet mounted/Shoulder worn

The current project is a preliminary attempt at developing such devices indigenously; hence, the focus is on the static variant.

4 Methods and Algorithms

4.1 Acoustic Signature of Gunshots

4.1.1 Muzzle blast from propellant at the gun location

A conventional firearm uses an explosive propellant in its muzzle to discharge the bullet. The sound from this explosion travels in all directions at sonic speed. Although the speed of propagation of the blast wave is uniform in all directions, the amplitude is loudest in the direction of firing as represented in the simplified directivity plot Figure 1

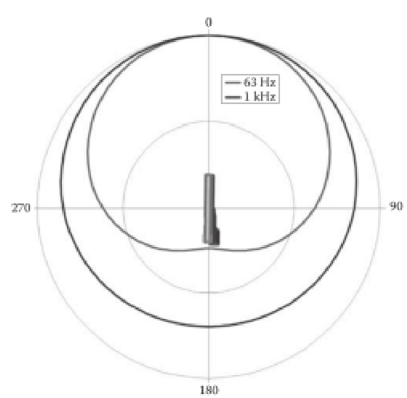


Figure 1: Simplification of muzzle blast Directivity (*J. R. Aguilar. Acoustic sensors and algorithms for urban security* [1])

The blast wave also suffers attenuation and broadening due to atmospheric absorption, wind, ground effects, and hindrance from solid obstacles. Every firearm has a distinct muzzle blast signature, as evidenced by Figure 2.

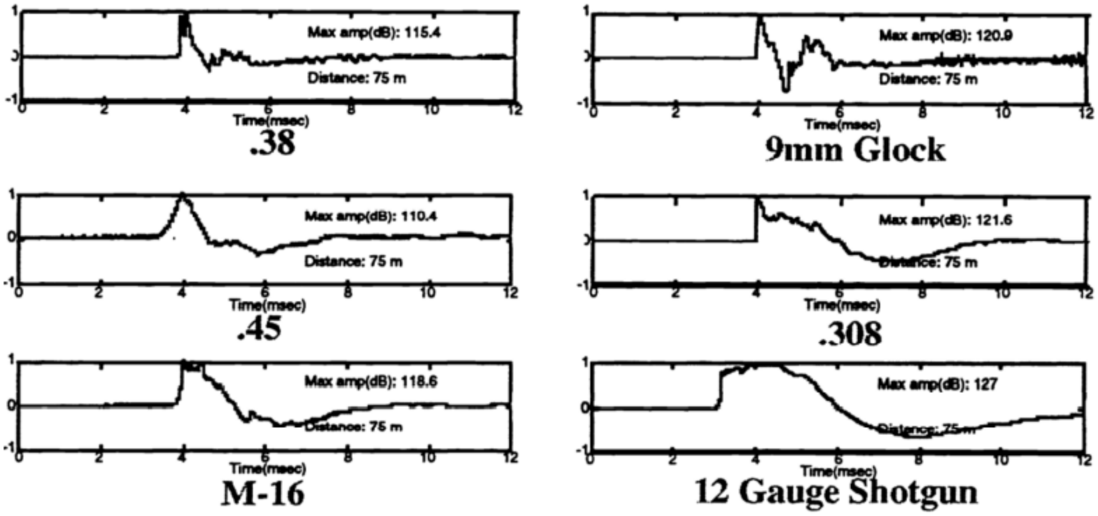


Figure 2: Forward direction muzzle blast waveforms for a variety of weapons (adapted from Page and Sharkey (1995) [2])

4.1.2 Shock waves generated by supersonically-moving projectiles

If the bullet travels faster than the speed of sound then it trails a shock wave in the form of a cone behind it. The leading edge of the bullet suddenly compresses the air in front of it and its trailing edge creates a corresponding expansion. Hence an 'N-wave' of shock is developed as described in Figure 3.

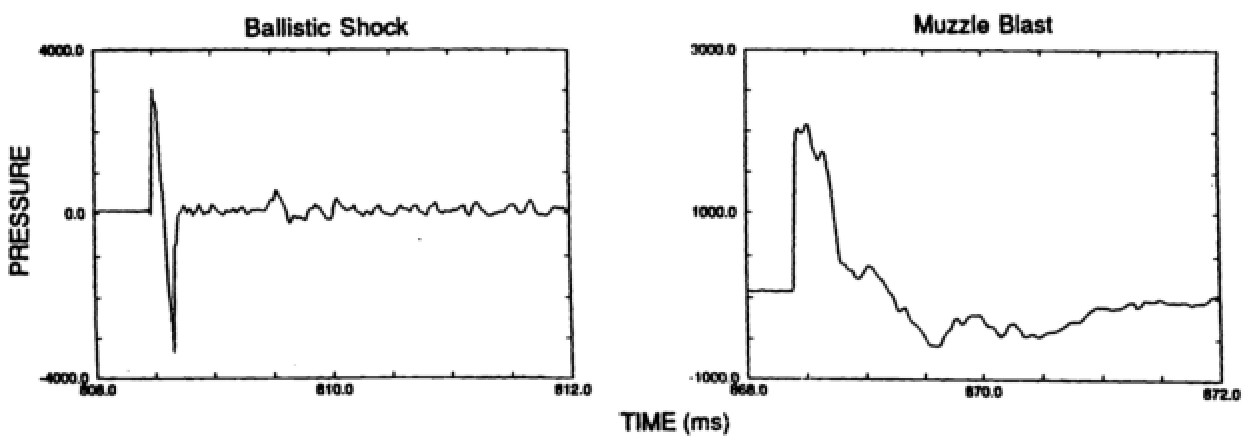


Figure 3: Ballistic shock and muzzle blast wave recorded from M-16 rifle. (Stoughton, 1997 [3])

The duration of this pulse is related to the calibre of the bullet. The amplitude is primarily a function of the 'miss distance' – how far away from the microphone the bullet passes.

4.2 Common Algorithms Description

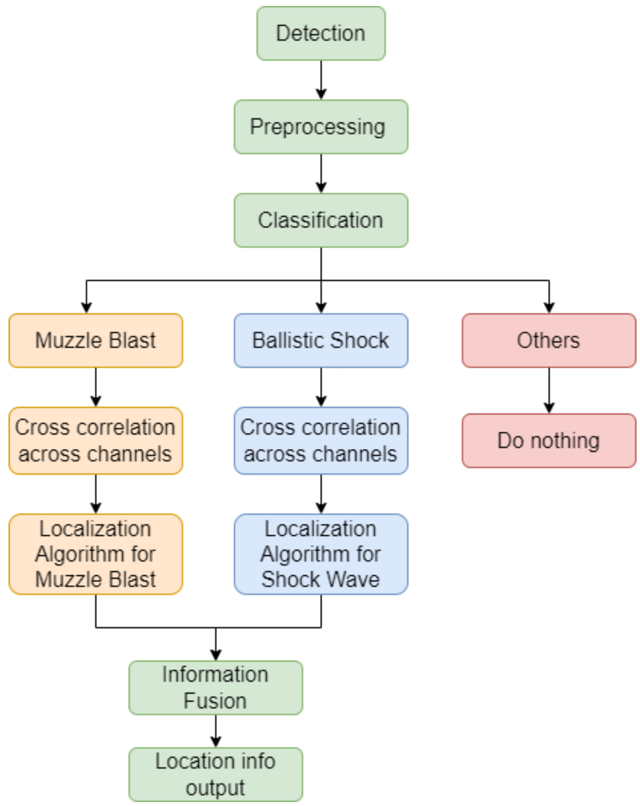


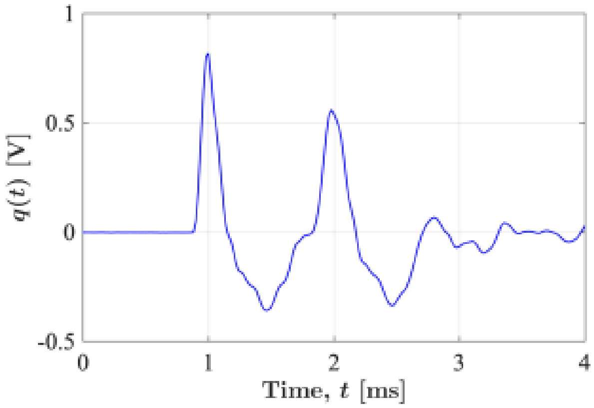
Figure 4: Algorithm overview for complete Gunshot Localization

4.2.1 Detection

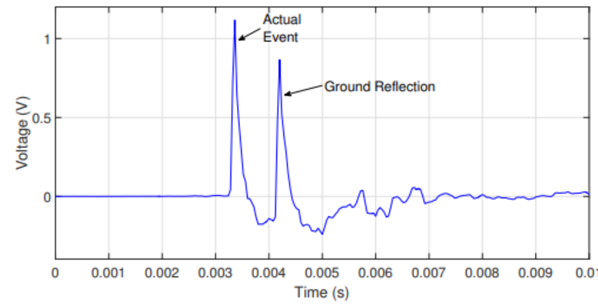
Preliminary detection of an acoustic event of interest is usually done by setting a threshold for the amplitude of the microphone signal. The threshold value depends on the microphone used, the approximate range of the shooter, the sound that the gunshot produces and the amplifying/shifting circuit post microphone.
Once the threshold is reached, we start recording the data. We save a few samples before the threshold is attained and a fixed number of samples after the threshold in order to capture the entire event on all the microphones.

4.2.2 Reflection Removal

The major reason for human confusion regarding the origin of an acoustic event (like gunfire) is echo from the ground and other reflectors.
Figure 5a exemplifies the issue with an audio recording of the firing event of a pistol. Due to reflection from the ground, a delayed and attenuated replica of the muzzle blast is recorded at the microphone.



(a) The audio recording of a Glock pistol's muzzle blast, with ground reflection



(b) The audio recording of a firecracker, with ground reflection

Figure 5: Effect of Reflection on Acoustic Signature [5]

Thus, if one were to know the differential time of arrival (DTOA) between the main event and its reflection as well as the attenuation factor, then one may be able to minimize the reflection artifact significantly (Braasch, 2013 [4]).

Cross-Correlation:
The cross-correlation between two sampled signals $q[n]$ and $r[n]$ shifted by k samples is given by:

$$R_{qr}[k] = \sum_{i=max(0,k)}^{N-1-min(0,-k)} q[i+k]r[i] \tag{1}$$

That is, the cross-correlation of two sequences q and r at a shift of k samples is the sum of the sample-wise product of the two sequences with the first one shifted ahead by k samples. The auto-correlation is obtained when both input sequences to the cross-correlation are the same.
The cross-correlation coefficient is given by:

$$\rho_{qr}[k] = \frac{R_{qr}[k]}{\sqrt{R_{qq}[0]R_{rr}[0]}} \tag{2}$$

It will be noted that $\sqrt{R_{qq}[0]}$ and $\sqrt{R_{rr}[0]}$ are simply the root mean square (RMS) of the q and r sequences, respectively. The cross-correlation is bounded between +1 and 1.
Thus, the cross-correlation coefficient gives the degree of similarity between two sequences, at all possible time shifts.

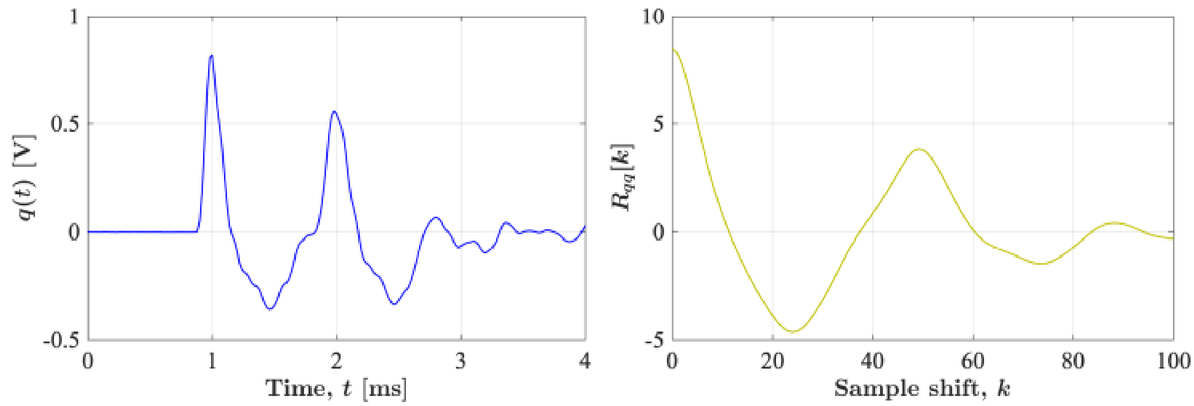


Figure 6: Autocorrelation of Muzzle blast signature of a Glock pistol with reflection artifact

The reflection artifact appears as the second peak in the auto-correlation of the sequence shown in Figure 6
The ratio γ of the second peak of R_{qq} to the peak at zero shift, was shown by Braasch (2013) [4] to be related to the attenuation: ratio α as:

$$\alpha = 0.5(1 - \sqrt{1 - 4\gamma^2})/\gamma \tag{3}$$

Let k^* denote the location of the second highest peak in the autocorrelation.
If we we delay the original sequence $q[i]$ by k^* and scale it down by the factor α, the main event will almost match the first reflection artifact in the original signal. Subtraction of the shifted and scaled sequence $\alpha q[i - k^*]$ from the original sequence yields a result that has significantly reduced reflection artifact. This however creates an inverted effect later in the signal which can further be reduced by adding the sequence $\alpha^2 q[i - 2k^*]$ to the first result.
The general formula for applying M corrections is:

$$\hat{q}[i] = \sum_{j=0}^{M} (-\alpha)^j q[i - jk^*] \tag{4}$$

Usually, M = 2 or 3 corrections are enough for the purposes of cleaning up the gunshot acoustic event sequence sufficiently.

4.2.3 Event Categorization

The muzzle blast and the bullet shock wave have fundamentally different propagative properties, and hence must be processed differently for shooter localization.
From Figure 3 we can clearly see the different characteristic signal of the muzzle blast and the shock wave.
The following modified 'Friedlander wave' is a very good fit for the muzzle blast signatures:

$$f_{Friedlander}(t) = \begin{cases} 0 & t \leq t_0 \\ A(t - t_0)/t_r & t_0 < t \leq t_r \\ A[1 - (t - t_0 - t_r)/t_d]e^{-(t - t_0 - t_r)/t_d} & t > t_0 + t_r \end{cases} \tag{5}$$

Here, A is peak amplitude of the event signature, t_0 time, and t_d characterizes its subsequent decay time. The function are only four parameters to be determined by a nonlinear least squares fitting exercise. A common simplification is to neglect the rise time altogether (Beck et al., 2011) [6], which reduces the

number of free parameters to three.
For the ballistic shocks, the N wave given by the following equations proves to be an excellent representation [7]:

$$f_{\text{N-wave}}(t) = \begin{cases} 0, & t \leq T_0 \\ B(t - T_0)/T_r, & T_0 < t \leq T_0 + T_r \\ B[1 - 2(t - T_0 - T_r)/T_d], & T_0 + T_r < t < T_0 + T_r + T_d \\ B[(t - T_0 - T_r - T_d)/T_r - 1], & T_0 + T_r + T_d < t < T_0 + 2T_r + T_d \\ 0, & t > T_0 + 2T_r + T_d \end{cases} \tag{6}$$

Here, B is the amplitude of the peak (both positive and negative), T_0 is its time of arrival, T_r is the common rise times of the initial and final segments, and T_d is the signal 'duration' from the positive to the negative peak. If we neglect the two rising sections owing to their extreme rapidity, the number of free parameters come down to three.
The overall algorithm for event categorization goes as follows. We try to fit the event signature with both the Friedlander and the 'N' wave shapes. The better fit determines the category of the event. If neither fit is deemed good enough, then the event is not considered to be from a gunshot.
If both the events are present simultaneously in the given signal sample, the current algorithm cannot be trusted.

4.2.4 DTOA using cross-correlation

Owing to the spatial separation of the microphones in an array, the acoustic wave from the gunshot arrives with a slight delay at the farther microphone compared to the nearer one. Hence the two channel signals are essentially time-shifted and amplitude-scaled versions of each other.
The DTOA can be best estimated as the sample count $\hat{k^*}$ corresponding to the peak of the cross-correlation of the two channels' signals. This $\hat{k^*}$ is of course the number of samples by which one signal has to be shifted (advanced or delayed) to match best with the other signal.
At worse, such discretization error in the DTOA estimate can be /2. Hence, we use parabolic interpolation to obtain a refined estimate:

$$k^* = \hat{k}^* + \frac{1}{2} \frac{R\left[\hat{k}^* - 1\right] - R\left[\hat{k}^* + 1\right]}{R\left[\hat{k}^* - 1\right] - 2R\left[\hat{k}^*\right] + R\left[\hat{k}^* + 1\right]} \tag{7}$$

where $R[.]$ is the cross correlation function.
This is illustrated in Figure 7

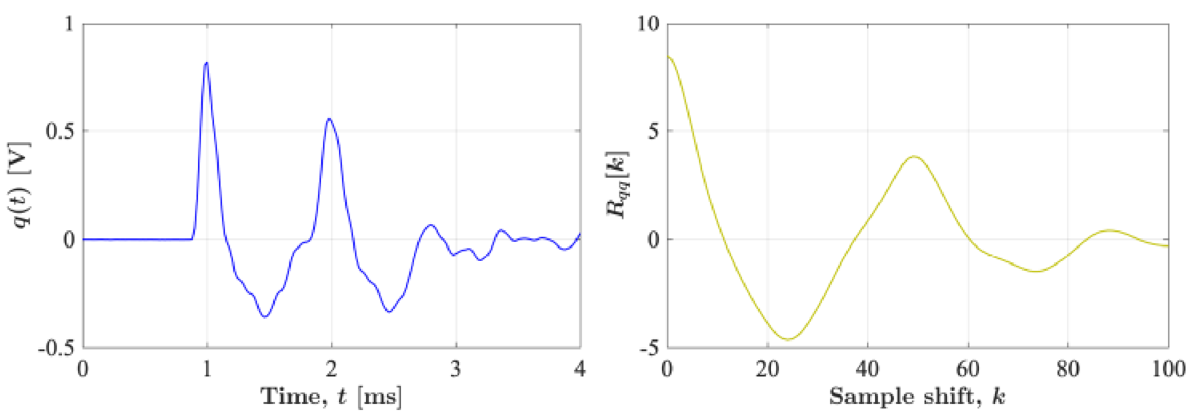


Figure 7: Parabolic interpolation of the Cross correlation Function

4.3 Algorithms for localizing gun-shooter from muzzle blast

the muzzle blast propagates approximately spherically in all directions from the gun muzzle. If one were able to identify any spherical wavefront of the blast, then the shooter may be directly located at its center. To define a spherical surface in three dimensions, one needs to know the coordinates of at least four points on it.
A significantly simpler scenario, which still happens to be almost equally relevant, is the corresponding two-dimensional problem. Here, we will have an array of at least three microphones in a plane. We will assume that the shooter is also in the same plane, so that the muzzle blast wavefront can be taken to be circular.
The spherical or circular muzzle blast wavefront also becomes very flat by the time it arrives at the sensor array, so that the calculation of its center from the DTOA measurements is very problematic. In other words, we expect significant inaccuracy in the range estimate of a shooter if we use a small array of sensors. But we can get the bearing angle of the gunshot with much greater precision.
Following are the algorithms we can use to estimate the shooter location using the muzzle blast.

4.3.1 Wavefront Curvature Method

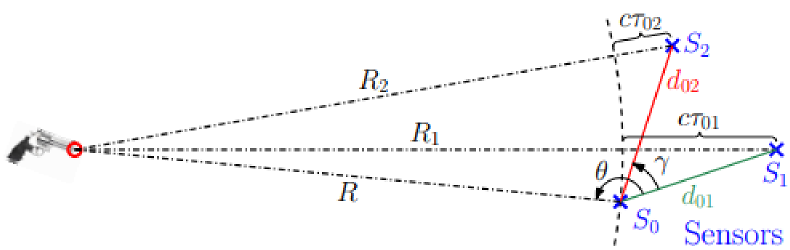


Figure 8: Schematic of the wavefront curvature method for 2-D localization of a gun-shooter

The wavefront curvature method was proposed by Carter (1981) [8] for locating a gunshot source using its muzzle blast wave. This is a planar localization technique, and thus assumes circular expansion of the acoustic wavefront. We need three minimum microphones for this purpose. The overall schematic of this method is shown in Figure 8. The wavefront curvature method seeks an exact solution to the problem. As shown in Figure 8, the sound waves propagating circularly from the source reaches the sensors $S0$, S_1 and S_2 through wavefronts of radius R, R_1 and R_2 respectively. Wedefine the DTOAs as $\tau_{ij} = t_j - t_i$, with t_k being the time of arrival of the wavefront at the sensor S_k.
From the geometry we see that:

$$R_1 = R + c\tau_{01} \tag{8}$$

Using the cosine formula:

$$\cos\theta = \frac{R^2 + d_{01}^2 - R_1^2}{2Rd_{01}} \tag{9}$$

we have, we get

$$R = \frac{d_{01}}{2}\frac{1-\eta_{01}^2}{\cos\theta + \eta_{01}} \tag{10}$$

(eliminating R_1 and some algebraic manipulations)
where η_{ij} is the normalized TDOA between sensor pair $S_i - S_j$ defined as:

$$\eta_{ij} := c\tau_{ij}/d_{ij} \tag{11}$$

A similar derivation for the triangle formed by S_0, S_2 and the source (with the included angle being $\theta-\gamma$) gives

$$R=\frac{d_{02}}{2} \frac{1-\eta_{02}^2}{\cos (\theta-\gamma)+\eta_{02}} \tag{12}$$

After eliminating the range R from the above two equations, bearing angle θ is obtained as

$$\theta=\tan ^{-1}\left(\frac{\sin \gamma}{\cos \gamma-\delta}\right) \pm \cos ^{-1}\left(\frac{\delta \eta_{01}-\eta_{02}}{\sqrt{1+\delta^2-2 \delta \cos \gamma}}\right), \quad \delta:=\frac{d_{02}\left(1-\eta_{02}^2\right)}{d_{01}\left(1-\eta_{01}^2\right)} \tag{13}$$

It is evident from the bearing angle estimation formula of Equation 13 that there are two mathematically-possible solutions delivered by the method. One solution invariably corresponds to a very nearby source location, whereas the other one is much farther away from the sensor array [9]. Since most practical cases will be of the latter type, one can safely ignore the nearby solution, and say that the farther one is the solution of the wavefront curvature method.

We obtain a closed-form solution for the location (bearing angle and range) of the gunshot source in the plane of the sensor arrangement for an exactly-determined system.

4.3.2 Non-linear Least square method

The wavefront curvature method is not directly applicable to the more robust case where we happen to have more sensors in an array than the strict minimum. In these general scenarios, the problem of localization may be formulated as an algebraic nonlinear least-squares problem, and solved using the classical Gauss-Newton method [10].

The muzzle blast propagation equations can be written down for M microphones as:

$$c t_i=D\left(\boldsymbol{x}_i, \boldsymbol{z}\right)+c T, \quad i \in\{0,1, \ldots, M-1\} \tag{14}$$

Here, T is the instant of gunfire at the position $\boldsymbol{z}$ (which is a column vector), t_i is the time-of-arrival (TOA) of the muzzle blast at the i th microphone located at $\boldsymbol{x}_i, c$ is the ambient speed of sound, and $D(\cdot, \cdot)$ is the distance function that takes as argument two position vectors and gives as output the distance between them.

Then, we obtain $M-1$ equations in terms of the differential times of arrival DTOAs.:

$$c \tau_{0 i}=D\left(\boldsymbol{x}_i, \boldsymbol{z}\right)-D\left(\boldsymbol{x}_0, \boldsymbol{z}\right), \quad i \in\{1, \ldots, M-1\} \tag{15}$$

Here we need to solve for z. Rest all the quantities are known.

Thus, we define the errors or 'residuals' of the $M-1$ nonlinear equations as:

$$r_i(\boldsymbol{z}):=c \tau_{0 i}-\left\{D\left(\boldsymbol{x}_i, \boldsymbol{z}\right)-D\left(\boldsymbol{x}_0, \boldsymbol{z}\right)\right\}, \quad i \in\{1, \ldots, M-1\} \tag{16}$$

The Gauss-Newton algorithm iteratively finds the value of z that minimizes the sum of the squares of the residuals:

$$S(\boldsymbol{z}):=\sum_{i=1}^{M-1}\left|r_i(\boldsymbol{z})\right|^2=\boldsymbol{r}(\boldsymbol{z})^{\mathrm{T}} \boldsymbol{r}(\boldsymbol{z}) \tag{17}$$

The algorithm requires an initial estimate $z^{(0)}$ which can come from the solution of wavefront curvature method. Then, the $(k+1)$ th iteration of the solution, $\mathbf{z}^{(\mathrm{k+d})}$, is given by:

$$z^{(k+1)}=z^{(k)}-\left(J\left(z^{(k)}\right)^{\mathrm{T}} J\left(z^{(k)}\right)\right)^{-1} d\left(z^{(k)}\right)^{\mathrm{T}} \mathrm{r}\left(\mathrm{z}^{(k)}\right) .$$

Here, $J\left(z^{(E)}\right)$ is the Jacobian matrix of the residual vector function $r(z)$ evaluated at the kth iterate of the solution $z=z^{(k)}$.

$$J_{i j}(\boldsymbol{z}):=\frac{\partial r_i}{\partial z_j}=\frac{x_{i j}-z_j}{D\left(\boldsymbol{x}_i, \boldsymbol{z}\right)}-\frac{x_{0 j}-z_j}{D\left(\boldsymbol{x}_0, \boldsymbol{z}\right)} \tag{18}$$

5 Implementation of Gunshot Localization on Hardware

It is not always easy to get access to live gunfire for testing and validation of our targeting device. Hence, the possibility of using some other sound source to simulate gunfire was explored. From the figure 5b, we can clearly see that fire-crackers also produce a very similar audio signature as compared to the gunfire. However, the firecrackers do not produce a shock wave and hence, to start with, the audio source localization using muzzle blast was implemented.

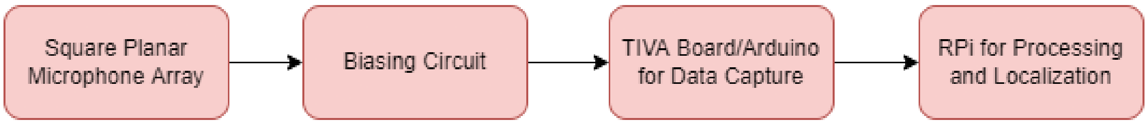


Figure 9: Hardware Schematic for Gunshot Localization

5.1 Proposal for the Choice of Hardware

5.1.1 Microphone Array

Sennheiser MD-42 microphones were chosen as the candidate microphones for the implementation as these are omni-directional dynamic microphones with sensitivity of 2 mV/Pa.

5.1.2 Biasing Circuit

The microphones output is within the range of -1.65V to +1.65V. However, the TIVA board won't be able to read the negative voltages. Hence, we require a signal conditioning circuit as to shift this range to 0V to 3V respectively.

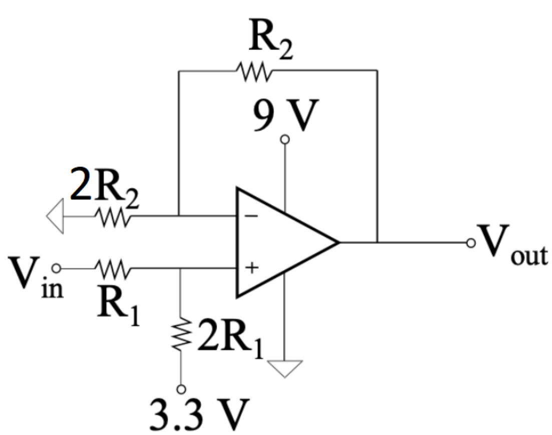


Figure 10: Biasing Circuit

From the equation of the op-amp, we know that:

$$V_+ = V_- \tag{19}$$

Equating currents we have,

$$\frac{V_+ - V_{in}}{R_1} = \frac{3.3 - V_+}{2R_1} \tag{20}$$

and

$$\frac{V_-}{2R_2} = \frac{V_{out} - V_-}{R_2} \tag{21}$$

Solving these equations we get,

$$V_{out} = 1.65 + Vin \tag{22}$$

5.1.3 TIVA Board for Data Acquisition

To implement the real-time localization algorithm, Tiva C Series TM4C1294 Connected LaunchPad Evaluation Board (Tiv) is chosen to acquire the microphone signals. It has an ARM-based microcontroller with two 12-bit ADC modules and Ethernet connectivity.
We use a sampling rate of 80 KSPS and a data buffer of 700 samples for each microphone channel to completely acquire the muzzle blast signature.
It does the following tasks:

- Continuously 'listen' on all microphone channels
- Detect the acoustic event on any microphone channel
- Acquire data in a brief time window (700 samples @ 80KSPS) covering the acoustic event on all four microphones, once an event is detected
- Pass on the collected data over Ethernet to a downstream microprocessor (RPi in our case)

5.1.4 RPi board for gunshot localization

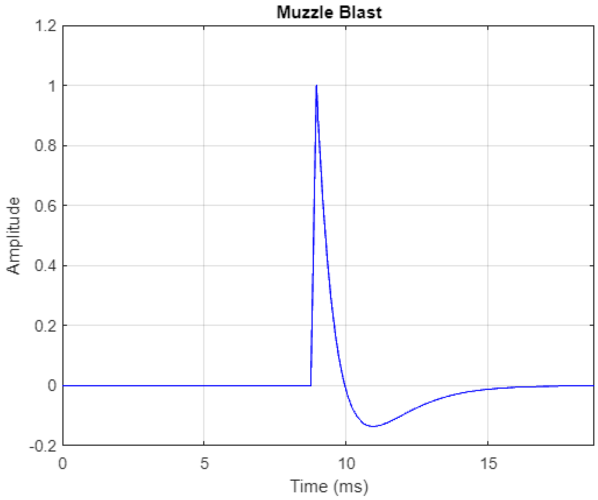
The actual acoustic localization algorithm requires computational power that the Tiva board cannot deliver. Thus, we propose to implement this on a Raspberry Pi Compute Module 3+ (RPi), which has a better processing capability.
On receiving the 4-microphone data through ethernet, the RPi, will calculate the position estimate using wavefront curvature method or non-linear least square method or both and display/send the results as required.

6 MATLAB Simulator for Data Generation

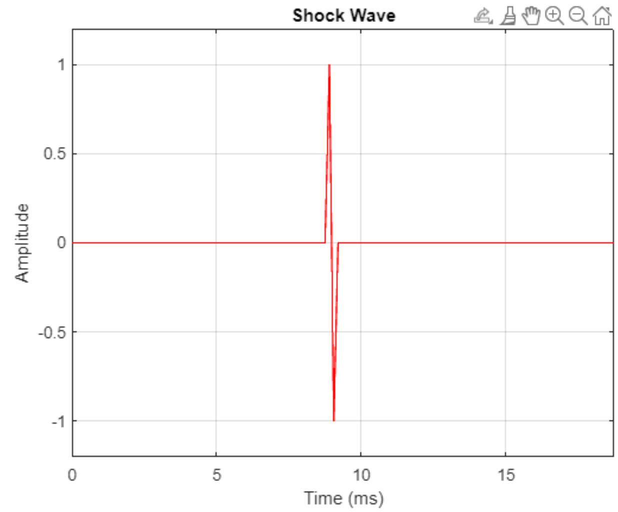
The main objective of this project was to implement the localization algorithm to find the bearing angle of the gunshot. We needed to generate the dummy microphone data in order to validate the algorithm. Hence a simulator was designed in MATLAB to simulate the gunshot and the microphone array and generate the 4-channel microphone data.

6.1 Generation of Muzzle Blast and Shock Wave Signatures

Using the equations of the muzzle blast and Shock Wave described in Section 4.2.3, the muzzle blast and N-wave data was generated. The different parameters were tuned to match the generated data with the actual data collected from the gunshot and presented in [5].



(a) Simulated Muzzle Blast



(b) Simulated Shock Wave

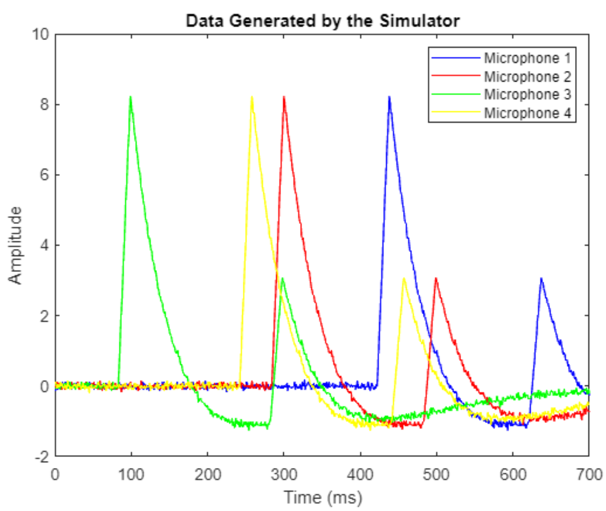
Figure 11: Effect of Attenuation due to Spherical Spreading

6.2 Modelling Path Loss in Air

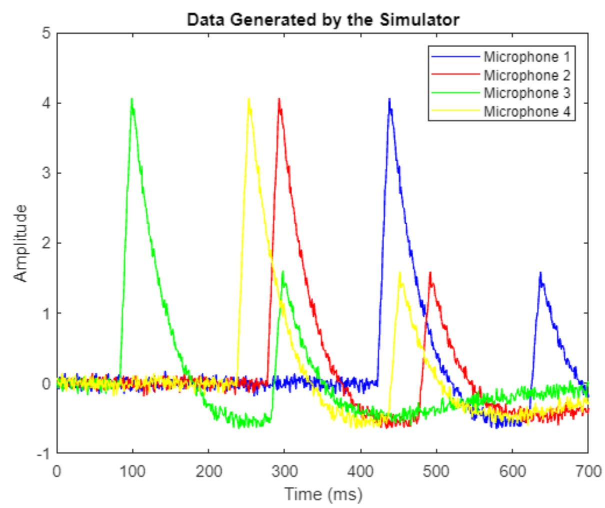
As the sound travels in air, it suffers an attenuation due to the spherical spreading of the wavefront governed by the equation:

$$\frac{P_r}{P_t} = D_t D_r \left(\frac{\lambda}{4\pi d}\right)^2 \tag{23}$$

where
- D_t is the directivity of the transmitting antenna
- D_r is the directivity of the receiving antenna
- λ is the signal wavelength
- d is the distance between the antennas

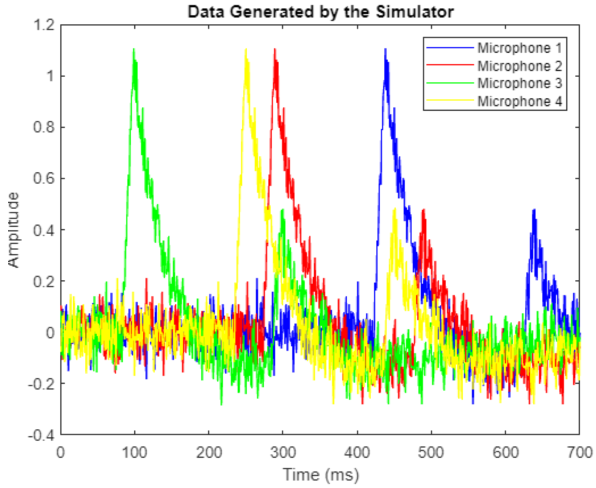


(a) Distance from array = 6.4 m

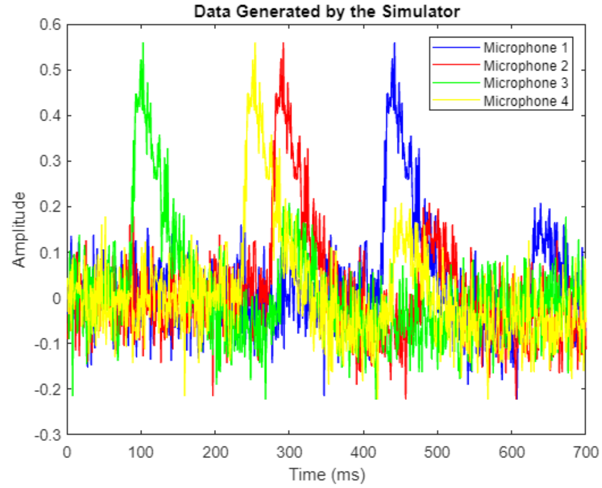


(b) Distance from array = 12.8 m

Figure 12: Effect of Attenuation due to Spherical Spreading



(a) Distance from array = 25.6 m



(b) Distance from array = 106.3 m

Figure 13: Effect of Attenuation due to Spherical Spreading

Observations:

- The noise levels of the received signals increase as the source moves away from the array.
- The SNR decreases with increasing range
- The attenuation of the signal causes the peak amplitude to decrease with increasing distance. For d=6.4m, peak amplitude = 8 units; for d=106.3m, peak amplitude = 0.58 units.
- After a distance of 160m, at noise power = -30dBW, the signal amplitude falls below 0.35 units and it is unable for our further classification algorithm to work properly. Hence the maximum range for the implementation at -30dBW noise is **160m**.

6.3 Addition of Reflection and Noise

White Guassian Noise with SNR as a tunable parameter was added to the above generated waveforms. We know that the reflected signal is a scaled and shifted version of the original signal. To model the ground reflection, the original signal is added with the reflected signal. The code also provides a functionality to add upto two reflections (can be scaled easily).

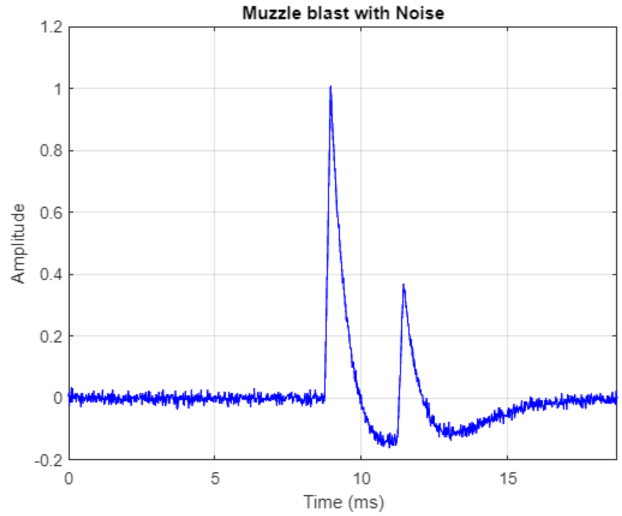


Figure 14: Simulated Reflections and Noise in Muzzle Blast

6.4 Modelling the Microphone Plane

To model the microphone plane and source, the simulator requires the following parameters:

- Microphone coordinates
- Coordinates of the gunshot source
- Speed of Sound in Air

Based on the geometrical calculations, a 4-dimensional array of 700 elements each is generated corresponding to the 4 microphones. This data is saved in a *.mat* file which is later passed to the RPi for getting the location estimate.

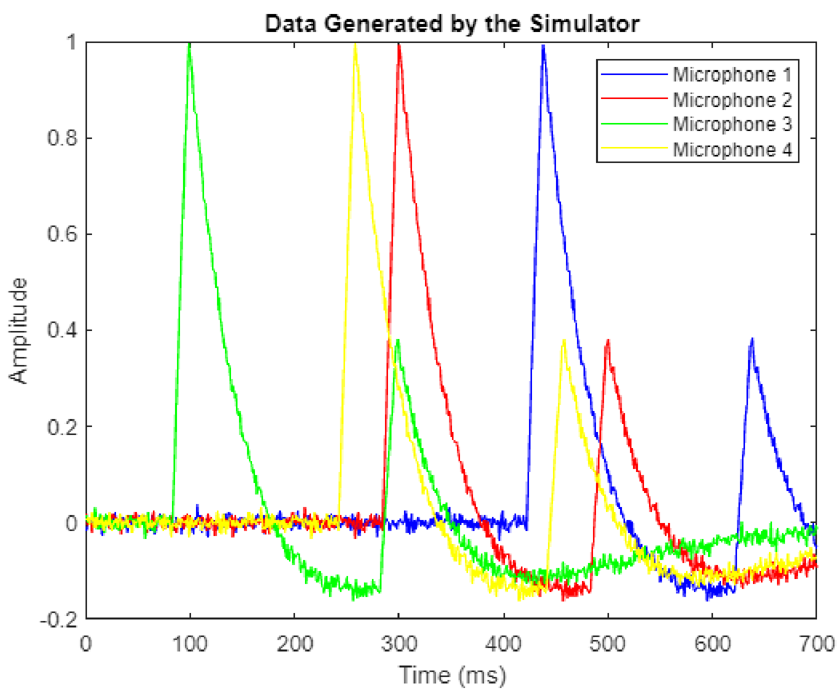


Figure 15: Simulated Data generated corresponding to 4-microphone array

7 RPi Implementation of Gunshot Localization

The code for Gunshot Localization is written in Python and is divided into 5 files:

- main.py
- preprocess.py
- reflection.py
- event_categorize.py
- WCM.py

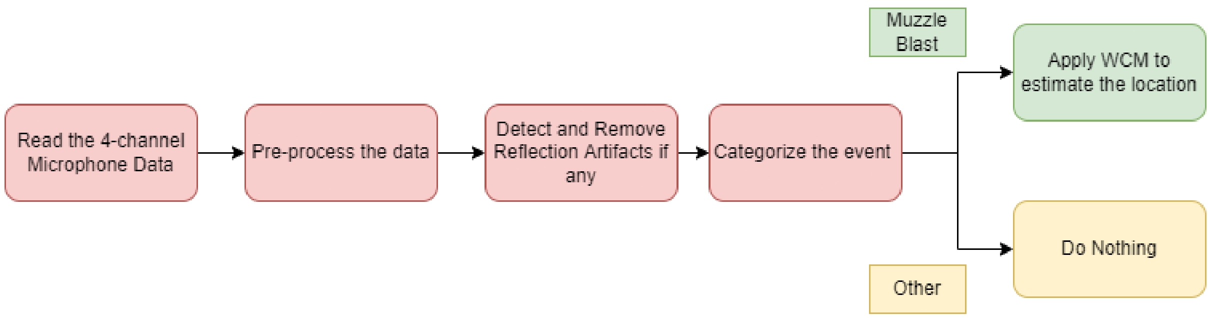
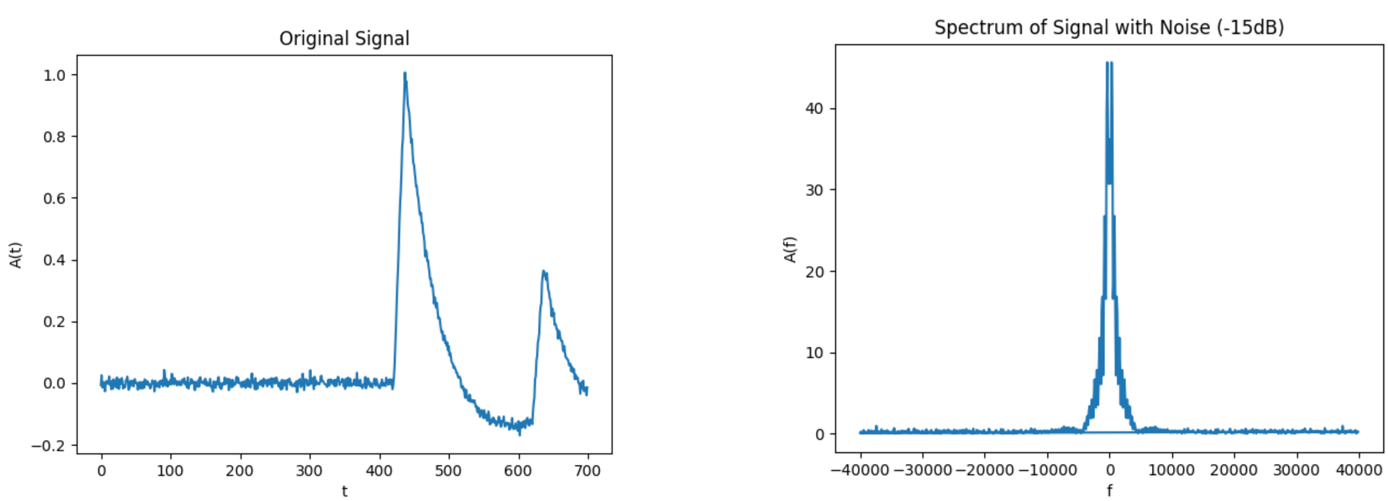


Figure 16: Process Flow for Gunshot Localization

7.1 Preprocessing the data

The incoming data might be noisy due to the surroundings and the ambient noise sources. Following is the raw data of muzzle blast (simulated with -15dB white noise) and its frequency spectrum:



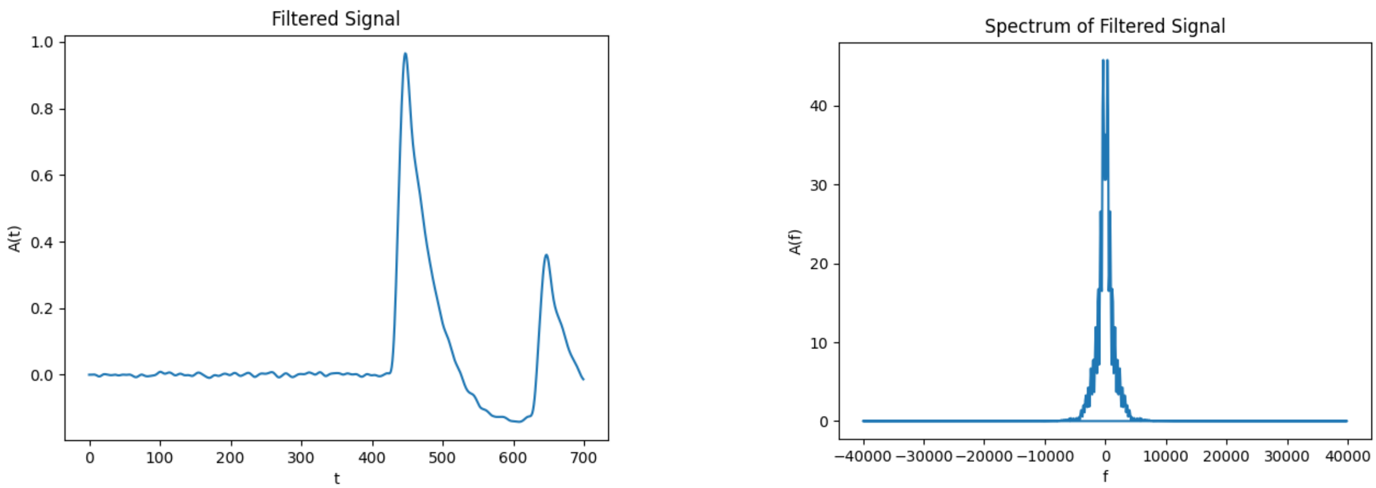
(a) Original signal captured by microphone

(b) Frequency Spectrum of the signal

Figure 17: Data captured by microphone

We see that the spectrum is mostly concentrated within 5000Hz and the frequencies beyond that would be mostly noise. So a butterworth lowpass filter of order 5 and 5000Hz as the cut-off frequency

was applied on this signal.
The *noise_remove* function in the *preprocess.py* file applies a low-pass filter to the incoming data and returns the filtered output.



(a) Time Domain

(b) Frequency Spectrum

Figure 18: Low-Pass Filtered Signal

Comparing the above 4 plots, we see that the noise was removed, however, the initial sharp rise of the signal was flattened and hence compromised due to the filtering.

7.2 Reflection Removal

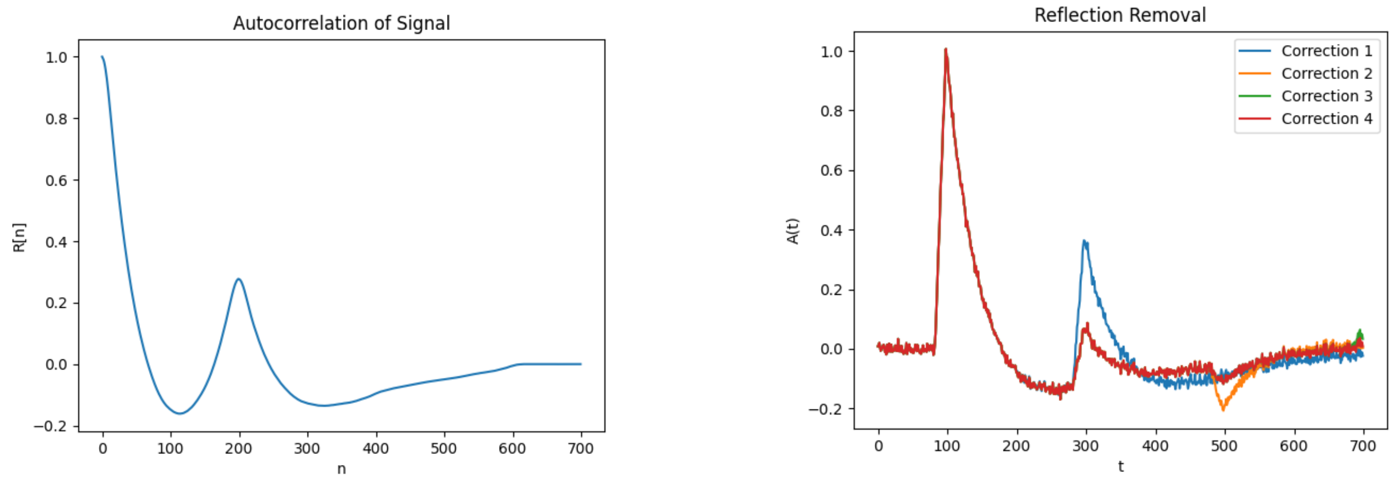
The process of reflection removal was as described in Section 4.2.2.

7.2.1 Detecting Reflection

The *detect_reflection* function in the *reflection.py* file detects if there is a reflection artifact present in the signal. This is done by finding the autocorrelation of the signal and detecting the second peak (other than that at 0) in the autocorrelation. If the peak is present, the location of the peak and the ratio to the peak at 0 is returned by the function.
This is done for all the microphone signals individually.
If no reflection is present, the function returns $None, None$.

7.2.2 Removing Reflection Artifacts

If a reflection is present in the signal, the *correct_reflection* function is called for that signal. The parameter $MaxCorrIter$ is the number of corrections to be applied as described in Section 4.2.2.
Following is an example which illustrates the working of the reflection removal code:



(a) Autocorrelation of Input Signal

(b) Reflection Corrections

Figure 19: Demonstration of 4-step Reflection Removal

In the figure 19b, we can clearly see that after applying 1 correction (Yellow plot), there are inverted signal artifacts generated, which is corrected by the further corrections (green and red respectively).

7.3 Event Categorization

Following is the algorithm followed for implementing event categorization in python:

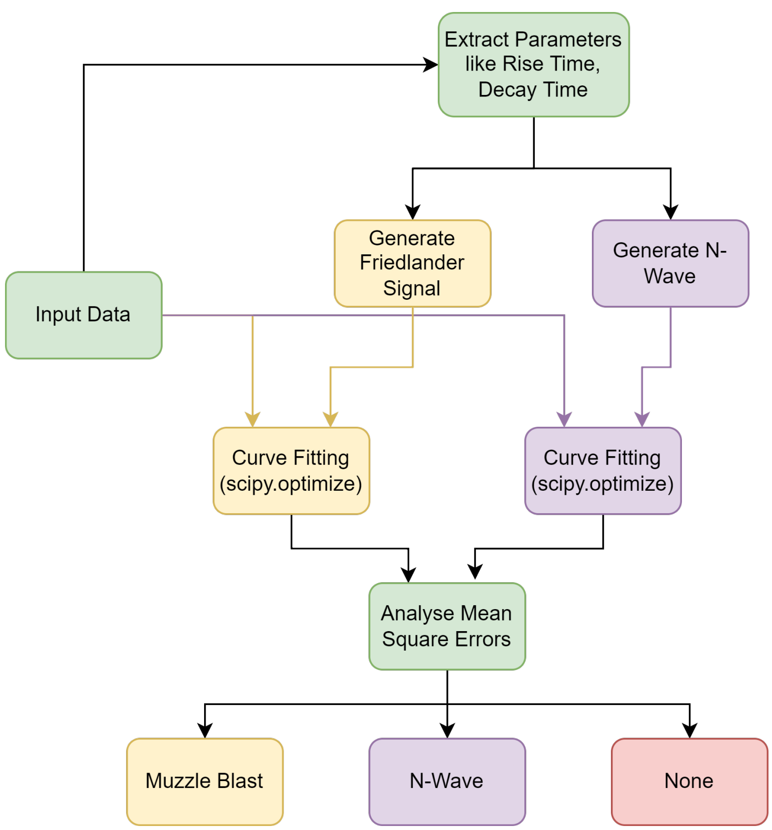
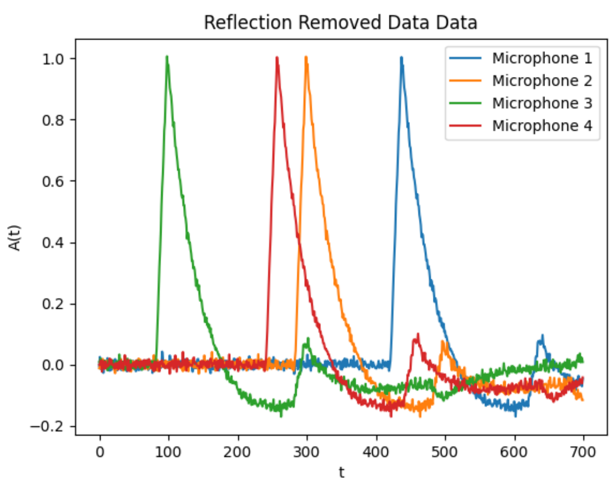


Figure 20: Event categorization algorithm

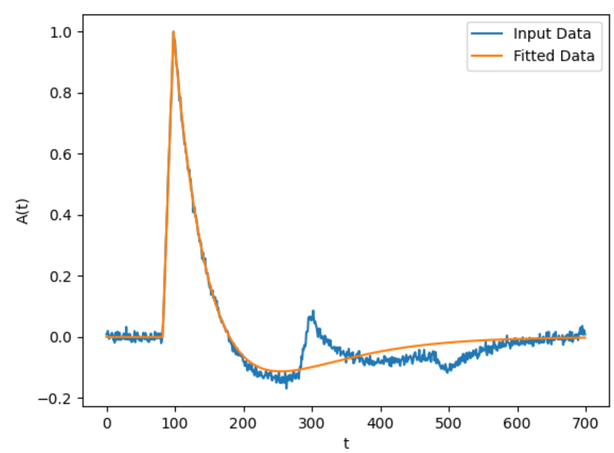
The file *event_categorize.py* contains 3 functions:

1. Friedlander
2. Nwave
3. categorize

The first 2 functions generate the ideal signals based on the input parameters are described in Section 4.2.3. The different signatures parameters are calculated from the input data and the *curve_fit* function from *scipy.optimize* library is used to fit the 2 curves to the input data.
Based on the mean square error of data fitting, the event is categorized into muzzle blast, shock wave or none.
Following is the demonstration of curve fitting for Muzzle Blast:



(a) Reflection Removed 4-channel Data



(b) Muzzle Blast fitting for Microphone 3

Figure 21: Demonstration of Data Fitting

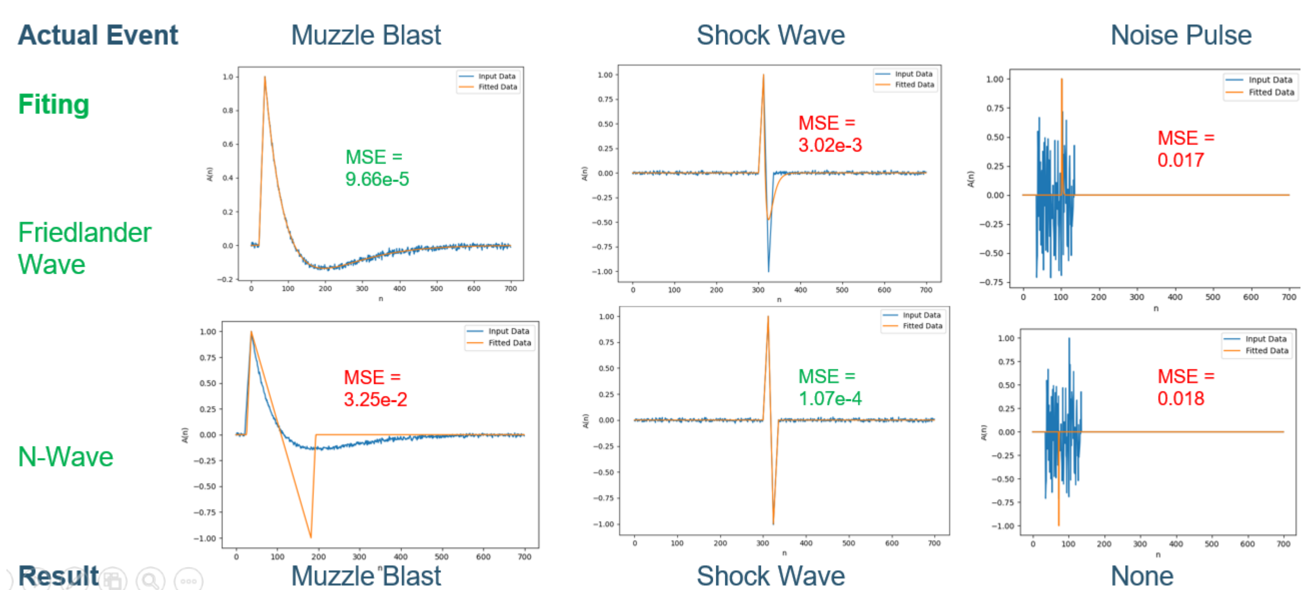


Figure 22: Fitting Friedlander and 'N' Waves for different events

In the figure 22, fitting of Friedlander and N-wave to 3 acoustic events namely - Muzzle blast, shock wave and noise pulse is demonstrated. The mean square error gives the goodness of fitting and acts as a tunable parameter for classification of the event. Based on the immense testing of simulated data, a MSE of ¡ 0.015 is considered as acceptable and then the fitting with lowest MSE is the classification result. If the all the errors ≥ 0.015, the event is classified as none.

7.4 Applying Wavefront Curvature Method

The *WCM.py* file implements the Wavefront Curvature Method as described in section 4.3.1. This is the exact geometrical solution obtained in terms of bearing angle (θ) and range R. First the DTOAs are calculated for each microphone pair and stored in the matrix *Delays*. This matrix is used to calculate the normalized TDOA and the bearing angle. The function *getValidSoln* validates the candidate solutions based on range calculation and selects the one with the higher range.

8 Results

8.1 Variation of Error with Source Location

The algorithm was tried was various test cases of Source Locations (X and Y) and the error in the estimated bearing angle was computed as follows:
(All Angles are in degrees)

Test Case	X	Y	Actual Angle	Estimated Angle	Error
muzzle_test_1	50	0	0.000	0.000	0.000
muzzle_test_2	50	10	11.310	11.024	0.286
muzzle_test_3	50	20	21.801	21.987	0.186
muzzle_test_4	40	30	36.870	36.994	0.124
muzzle_test_5	40	50	51.340	51.174	0.166
muzzle_test_6	40	40	45.000	45.000	0.000
muzzle_test_7	40	70	60.255	60.329	0.074
muzzle_test_8	0	50	90.000	90.215	0.215
muzzle_test_9	-20	40	116.565	116.424	0.141
muzzle_test_10	-40	40	135.000	134.830	0.170
muzzle_test_11	-60	0	180.000	180.000	0.000
muzzle_test_12	-50	-40	218.660	218.910	0.250
muzzle_test_13	0	-40	-90.000	-89.118	0.882
muzzle_test_14	40	-50	308.660	308.629	0.031

The RMSE error observed is 0.28°. This is quite low as expected because the data used is generated from the simulator.

8.2 Variation of Error with SNR

The added noise power to the input signal to the algorithm was varied from -30dBW to -10dBW for a fixed location of the sound source. The variation in the error of bearing angle with noise was observed as follows:

Noise (dbW)	X	Y	Actual Angle	Estimated Angle	Error
-50	-20	40	116.550	116.4357	0.114
-40	-20	40	116.550	116.4236	0.126
-35	-20	40	116.550	116.429	0.121
-30	-20	40	116.550	116.433	0.117
-25	-20	40	116.550	116.4361	0.114
-20	-20	40	116.550	116.4348	0.115
-15	-20	40	116.550	NAN	inf

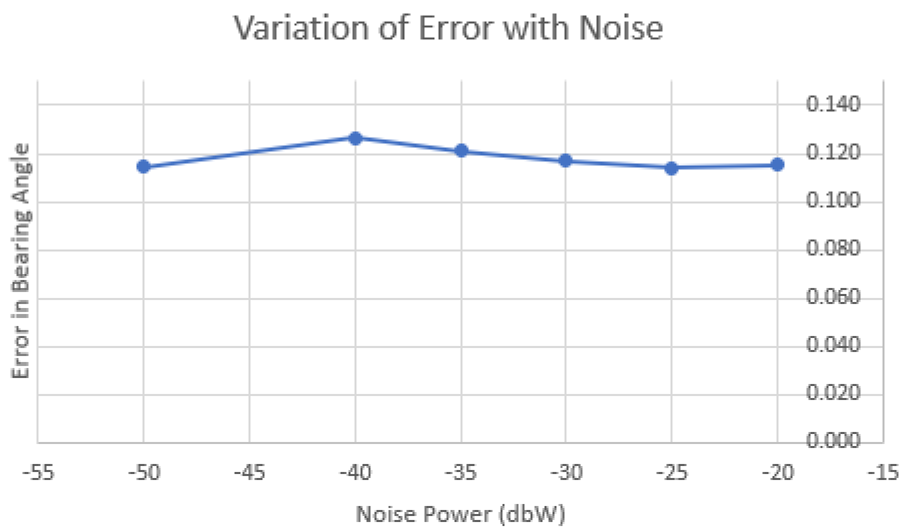


Figure 23: Variation of Bearing angle estimate with Noise Power

Figure 23 demonstrates that the performance of the localization algorithm is essentially independent of the noise to signal ratios investigated, within the indicated range. This behaviour can be understood by recalling the fact that cross-correlation is used to identify the DTOA for each pair of sensors in a cluster, and this is inherently robust to noise.

9 Conclusion

We see that the gunshot location can be found out using the various signal processing techniques as highlighted in the above report. There are two distinct acoustic signatures of a gunfire event – the spherically-expanding blast wave of the muzzle explosion, and the conical shock wave trailed by the bullet if it travels supersonically. Several muzzle blast localization algorithms have been surveyed in the project. Identifying the center of the wavefront is inherently difficult if the shooter is far from the sensor cluster because the curvature becomes very small. Although this impedes the accurate estimation of the range of the shooter, the bearing angle can still be predicted easily as it is given by the local propagation direction of the (almost planar) wavefront. A hardware was proposed was the implementation of the algorithms involving a TIVA board for sampling the data and RPi for further processing.
Finally a simulator was developed to mimic the gushot acoustic signatures and the microphone array. This data was used to develop and validate the localization algorithm involving wavefront curvature method.
The variation of error with different source locations and noise power was studied. We can also conclude that low-pass filtering is not required during the pre-processing step as the algorithms is almost immune

to the noise addition.
Hence, we see that the gunshot location can be found out using the various signal processing techniques as highlighted in the above report.
The further work in this project would involve to better the estimate using non-linear least square method. We can expand the scope of the code for multiple sensors placed at different locations. Finally we can also implement localization using shock wave and merge the two results to get a more accurate estimate.

References

[1] J. R. Aguilar. Acoustic sensors and algorithms for urban security. In J. R. Vacca, editor, Handbook of Sensor Networking: Advanced Technologies and Applications, pages 235–248. Chapman and Hall/CRC, 2015.

[2] A. Page and B. Sharkey. SECURES: system for reporting gunshots in urban environments. Proc. SPIE, 2497: 160–172, 1995

[3] R. B. Stoughton. SAIC SENTINEL acoustic counter-sniper system. Proc. SPIE, 2938:276–284, 1997

[4] J. Braasch. A precedence effect model to simulate localization dominance using an adaptive, stimulus parameter-based inhibition process. The Journal of the Acoustical Society of America, 134(1):420–435, 2013.

[5] Real-Time Gunshot Localization Using Acoustics, Aniruddha Sinha and Hemendra Arya, May 2021

[6] S. D. Beck, H. Nakasone, and K. W. Marr. Variations in recorded acoustic gunshot waveforms generated by small firearms. The Journal of the Acoustical Society of America, 129(4):1748–1759, 2011

[7] R. Maher. Modeling and signal processing of acoustic gunshot recordings. In 12th IEEE DSP Workshop 4th IEEE Signal Processing Education Workshop, pages 257–261, 2006.

[8] G. C. Carter. Time delay estimation for passive sonar signal processing. IEEE Transactions on Acoustics, Speech, and Signal Processing, 29(3):463–470, 1981.

[9] A. Kothiwala. Development of an algorithm for gun-shot localization. Master's thesis, Indian Institute of Technology Bombay, 2017

[10] N. Sirola. Closed-form algorithms in mobile positioning: Myths and misconceptions. In 7th Workshop on Positioning navigation and communication (WPNC), pages 38–44, 2010.