Real-Time Gunshot Localization Using Acoustics

EE451: Supervised Research Exposition

Presented by: Aditya Rajeev Harakare (18D100001)

Under the Guidance of: Prof. Rajbabu Velmurugan

Department of Electrical Engineering

Indian institute of technology Bombay

Project Objectives

- Study the **acoustic signatures** of a gunshot event
- Survey the different **algorithms** used for gunshot localization
- Existing **Hardware** for implementing the solution
- Design a **simulator in MATLAB** for simulating the muzzle blast from the gunshot and the planar 4-microphone array
- **Develop** a **Python code** for gunshot localization using Wavefront Curvature Method and test it on **RPi** using the data from the simulator
- Study the effect of **Noise and reflection artifacts** on the location estimate

Introduction

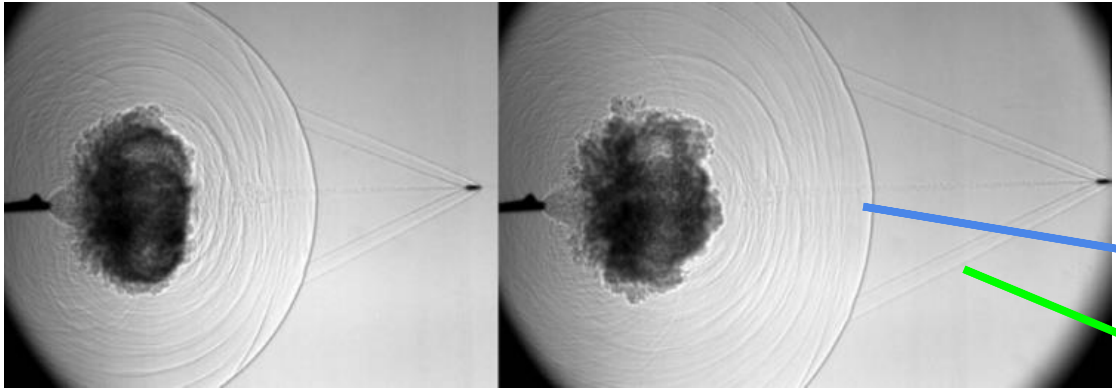
Signatures left by a Gunfire

- ✔ Sound of charge explosion in gun muzzle,
- Light of the charge explosion,
- ✔ Sound of the bullet travelling, which is only perceptible if the bullet's speed is supersonic,
- 'Sight' of the bullet travelling,
- Sound of the bullet hitting the target
- Smell of the gunfire

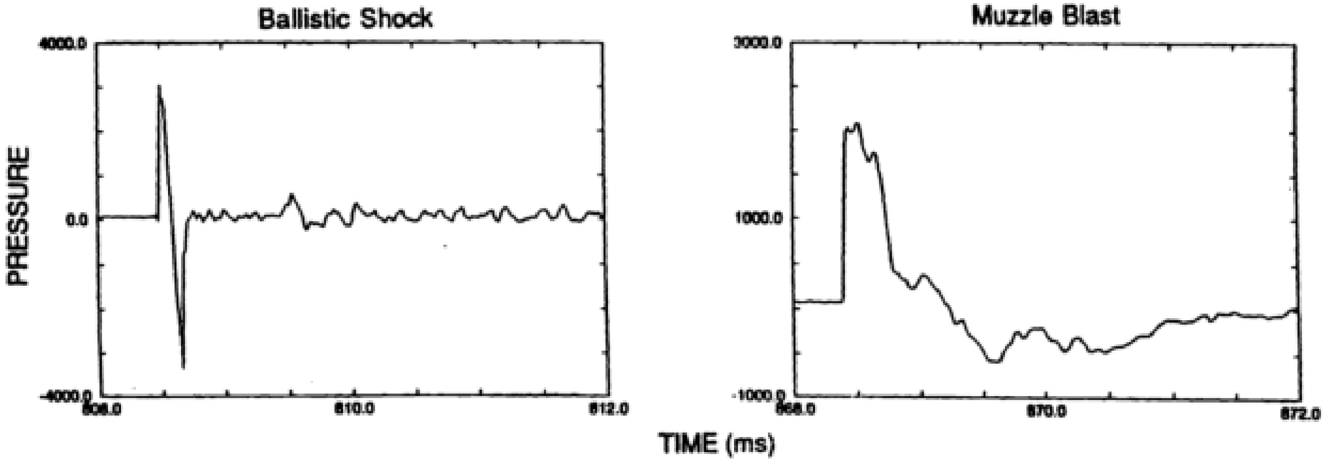
Classification of acoustic targeting devices based on mobility:

- ✔ Static Devices (city, forward camps in forests)
- Vehicle-mounted
- Helmet mounted or shoulder worn

Acoustic Signature of Gunshots



Consecutive shadowgrams of the firing of a Remington .30-06 deer rifle (Settles et al., 2005).

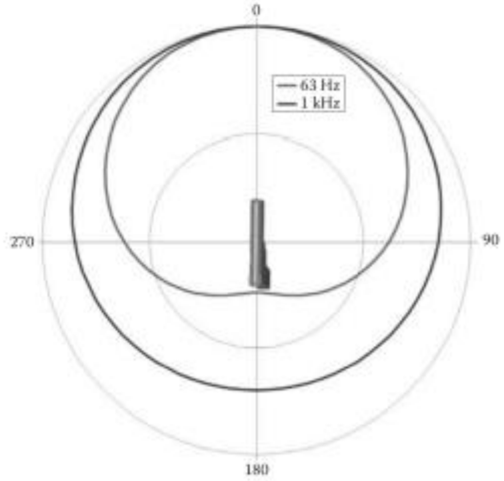


Two acoustic signatures:

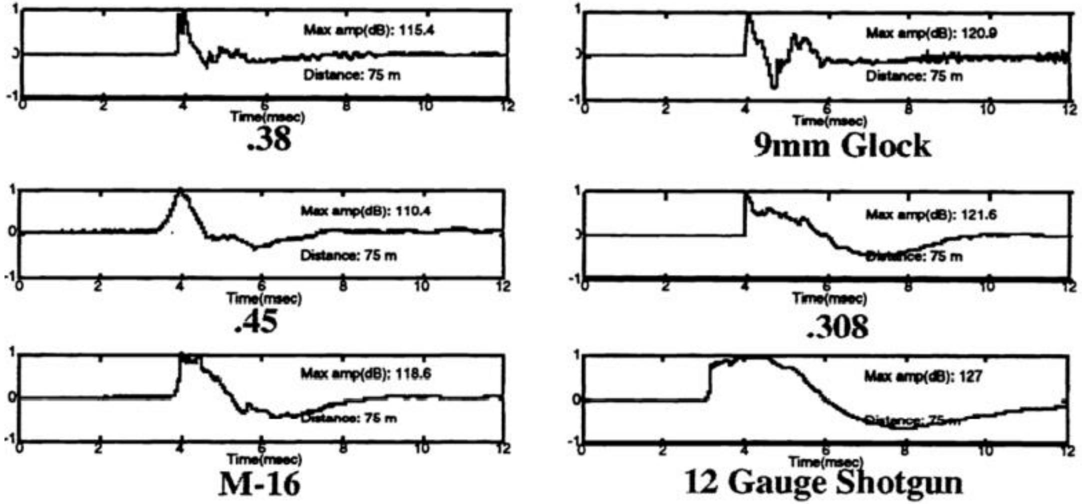
1. Muzzle Blast
2. Ballistic Shock Wave

Example pressure time series of ballistic shock wave and gun blast wave recorded from an M-16 rifle. (Stoughton, 1997).)

Acoustic Signature of Gunshots



Simplification of muzzle blast Directivity *(J. R. Aguilar. Acoustic sensors and algorithms for urban security)*



Forward direction muzzle blast waveforms for a variety of weapons (adapted from Page and Sharkey (1995)

Largest Amplitude in the direction of firing

Variation of signature for different guns; however similar shape

Acoustic Signatures left by a Gunshot

Muzzle Blast Signature

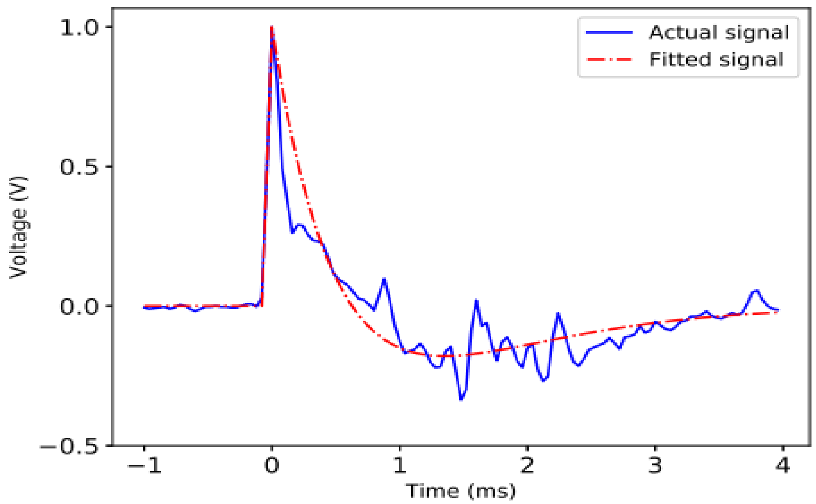
$$f_{Friedlander}(t) = \begin{cases} 0 & t \le t_0 \\ A(t-t_0)/t_r & t_0 < t \le t_r \\ A[1-(t-t_0-t_r)/t_d]e^{-(t-t_0-t_r)/t_d} & t > t_0 + t_r \end{cases}$$

A = peak amplitude

t_r = Rise Time (Can be neglected)

t_0 = TOA

t_d = Decay time



'N' Wave Signature (Shockwave)

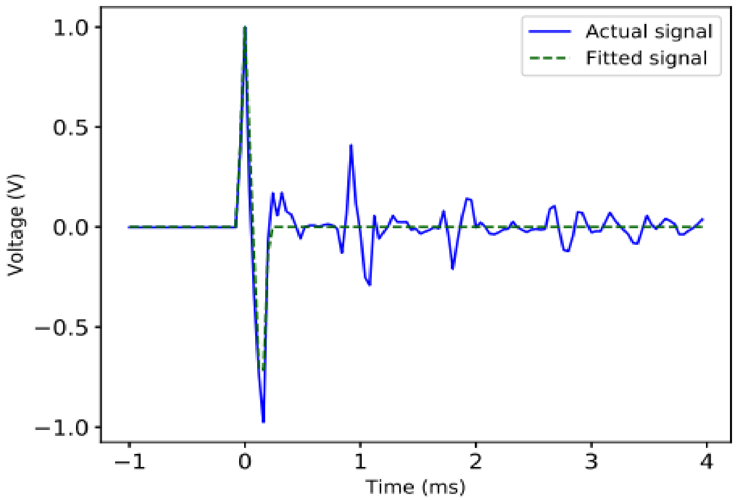
$$f_{\text{N-wave}}(t) = \begin{cases} 0, & t \le T_0 \\ B\,(t-T_0)\,/T_r, & T_0 < t \le T_0 + T_r \\ B\left[1 - 2\,(t-T_0-T_r)\,/T_d\right], & T_0 + T_r < t < T_0 + T_r + T_d \\ B\left[(t-T_0-T_r-T_d)\,/T_r - 1\right], & T_0 + T_r + T_d < t < T_0 + 2T_r + T_d \\ 0, & t > T_0 + 2T_r + T_d \end{cases}$$

B = peak amplitude

T_r = Rise Time (Can be neglected)

T_0 = TOA

T_d = Decay time



MATLAB Simulator for Data Generation

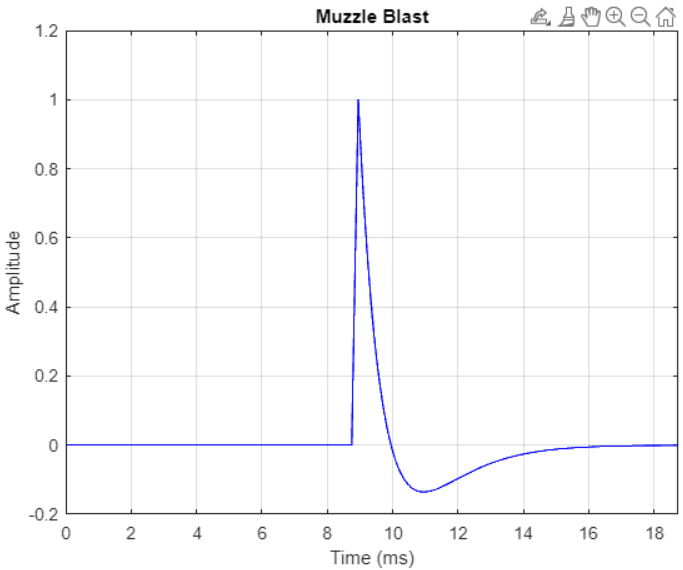
Need:

1. Non-availability of data from a gunshot

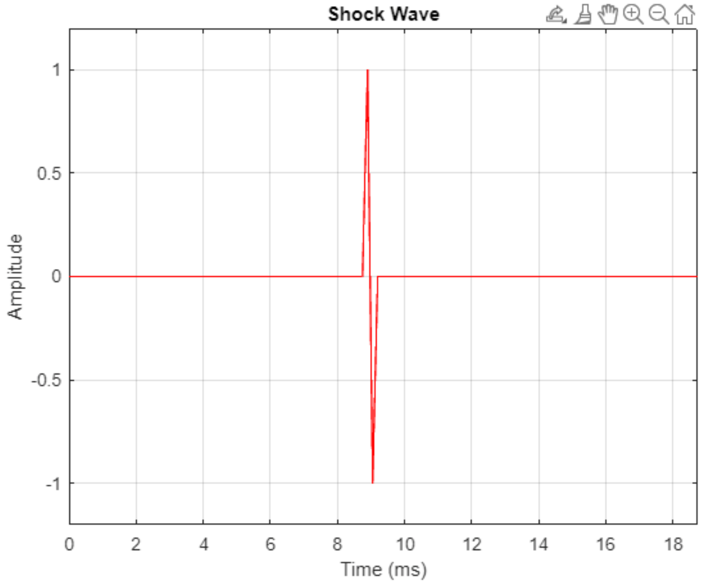
2. Validating and Testing of Localization Algorithms

Generation of Muzzle Blast and Shock Wave Signatures

Using the equations of the muzzle blast and Shock Wave described previously



Simulated Muzzle



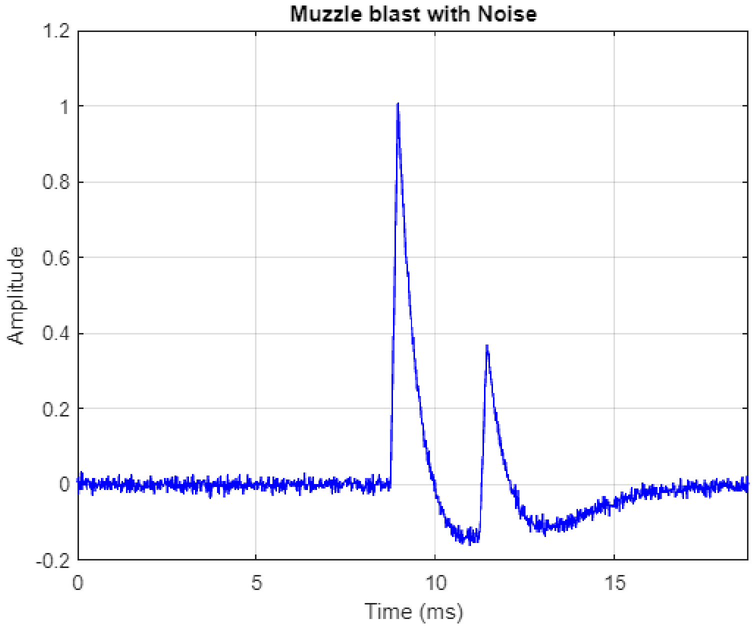
Simulated Shock Wave

MATLAB Simulator for Data Generation

Addition of Reflections and Noise

White Guassian Noise with SNR as a tunable parameter was added to the above generated waveforms.

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 up-to two reflections (can be scaled easily).



Simulated Reflections and Noise in Muzzle Blast

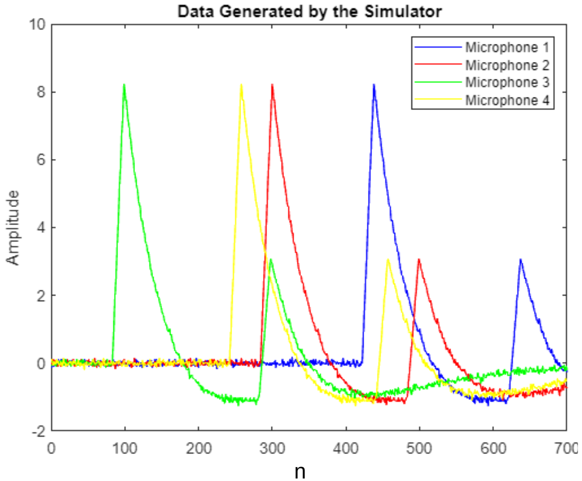
MATLAB Simulator for Data Generation

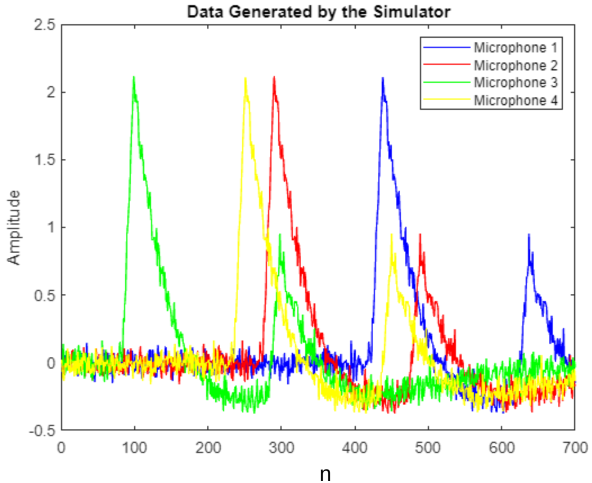
Modelling Path Loss in Air

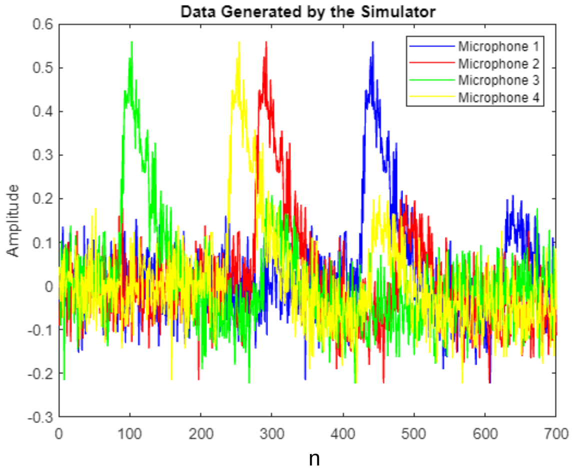
- As the sound travels in air, it suffers an attenuation due to the spherical spreading of the wavefront governed by the equation:

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

- $P_r = Received\ Power$ $\quad P_t = Transmitted\ Power$ $\quad \lambda = Wavelength\ of\ the\ signal$
  $D_r\ and\ D_t\ are\ the\ directivities$ $\quad d = Distance\ travelled$







Distance from mic array

6.4 m

25.6 m

106.3 m

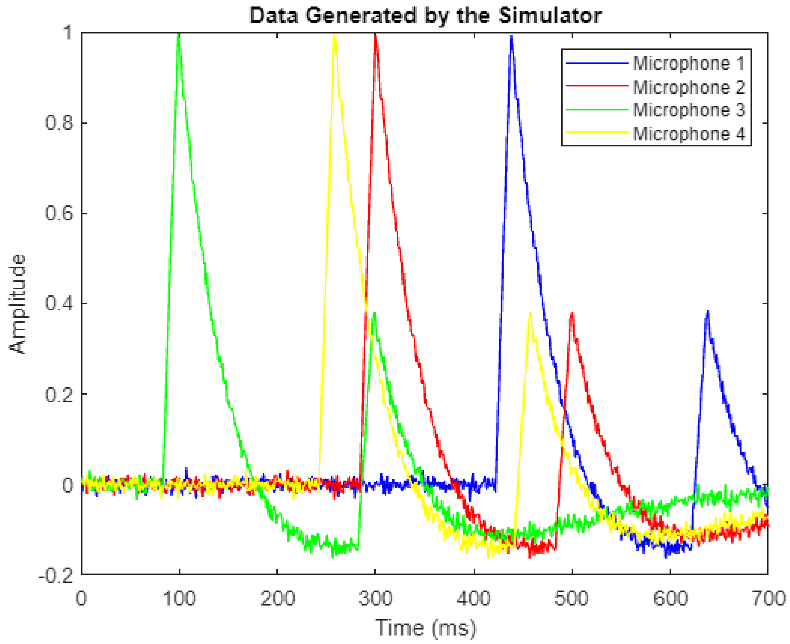
MATLAB Simulator for Data Generation

Modelling Microphone Plane

the simulator requires the following parameters:

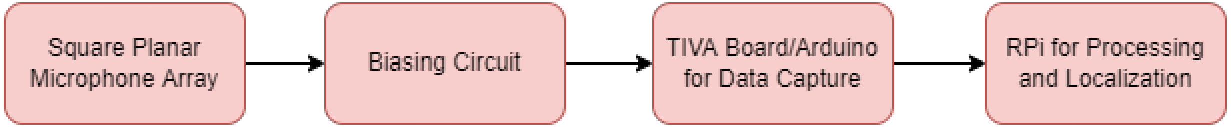
1. Microphone coordinates
2. Coordinates of the gunshot source
3. Speed of Sound in Air

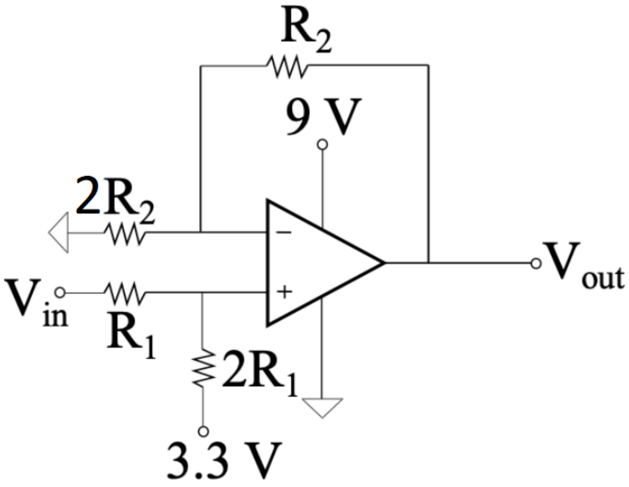
Based on the geometrical calculations, a 4-dimensional array of 700 elements each is generated corresponding to the 4 microphones.
This data is saved in a *.mat* file which is later passed to the **RPi** for getting the location estimate.



Simulated Data generated corresponding to 4-microphone array

Choice of Hardware for Localization



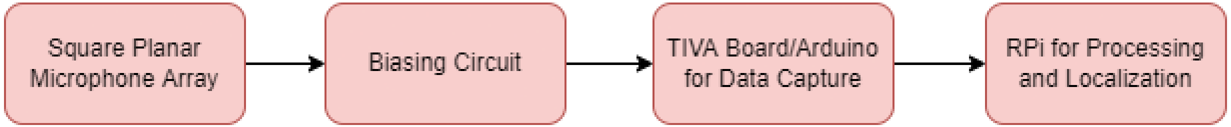


Sennheiser MD-42 microphones
sensitivity of 2 mV/Pa

Tetrahedron (3D)/Square (2D)
1m distance b/w mics

Biasing Circuit
Shifting Voltage range from
−1.65 to +1.65 V to 0 to 3V

Choice of Hardware for Localization



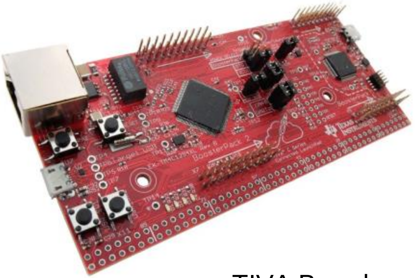
TIVA Board for Data Acquisition
Tiva C Series TM4C1294

80KSPS Sampling Rate
Data buffer of 700 samples for each microphone channel

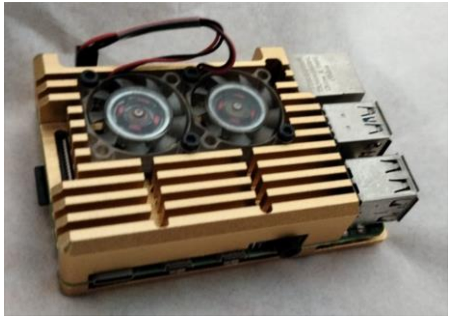
- Continuously ‘listen’ on all microphone channels
- Detect the acoustic event on any microphone channel
- Acquire data in a brief time window (700 samples @ 80KSPS) covering the acoustic event on all four microphones, once an event is detected
- Pass on the collected data over Ethernet to a downstream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

RPi board for gunshot localization
Raspberry Pi 4B

- Receives 4-channel microphone data through ethernet
- calculate the position estimate using wavefront curvature method or non-linear least square method or both
- Display/send the results as required

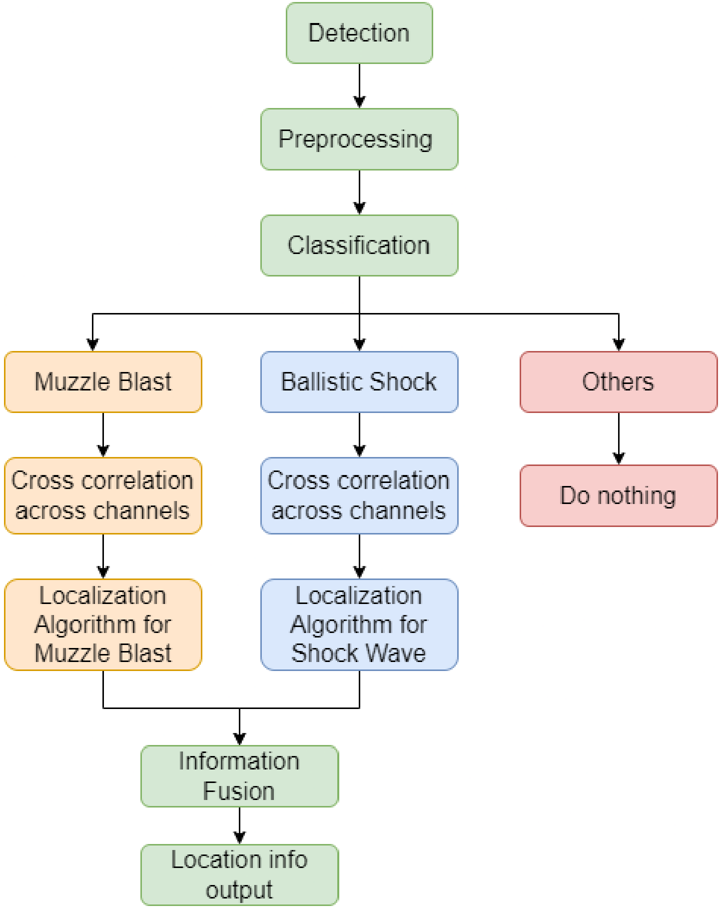


TIVA Board



RPi Board

Gunshot Localization Algorithm Flow

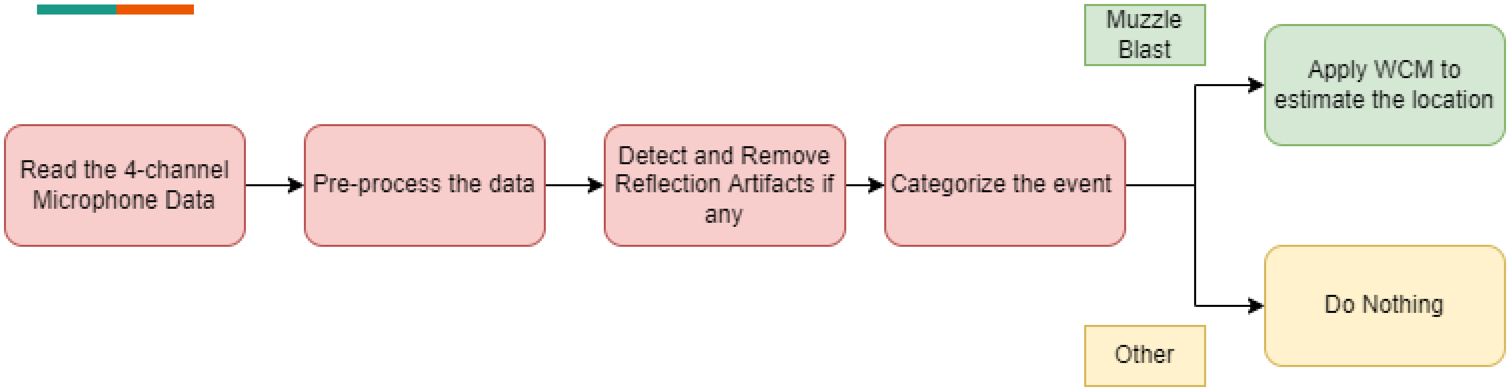


Detection (On Sampling Device)
Threshold for the amplitude of the microphone signal.
Once the threshold is reached, we start recording the data.

Pre-processing

1. Noise Removal
   - Low pass filtering
2. Reflection Removal

RPi Implementation of Gunshot Localization

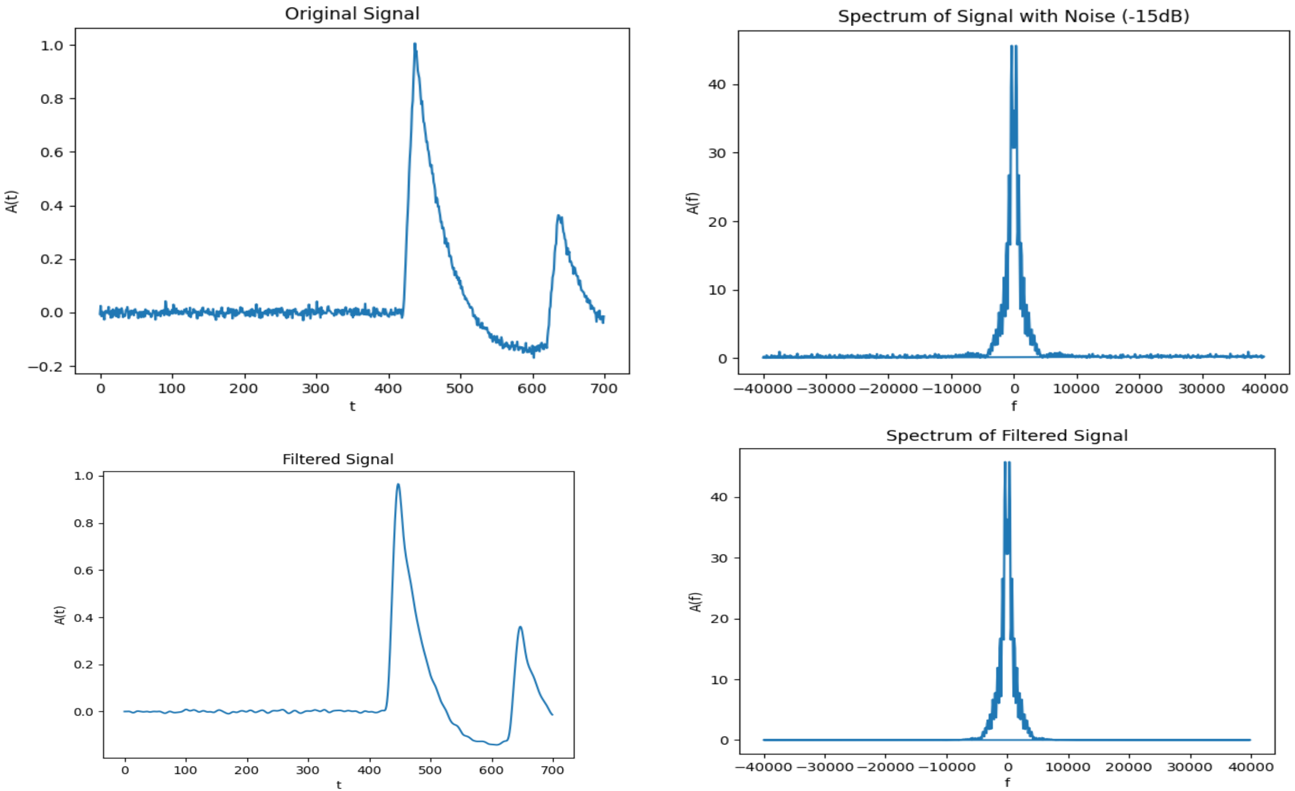


The code for Gunshot Localization is written in Python and is divided into 5 files

1. main.py
2. preprocess.py
3. reflection.py
4. event_categorize.py
5. WCM.py

RPi Implementation of Gunshot Localization

Preprocessing the Data (Noise Removal)



The incoming data might be noisy due to the surroundings and the ambient noise sources.

The top figures are the raw data of muzzle blast (simulated with -15dB white noise) and its frequency spectrum

After applying a 5th order low-pass filter with 5000 Hz as cut-off frequency, we get the signal as shown in the bottom plots

Reflection Removal

Due to reflection from the ground, a delayed and attenuated replica of the muzzle blast is recorded at the microphone

1. Find autocorrelation

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

2. Find

γ = Ratio of the second peak of Rqq to the peak at zero shift

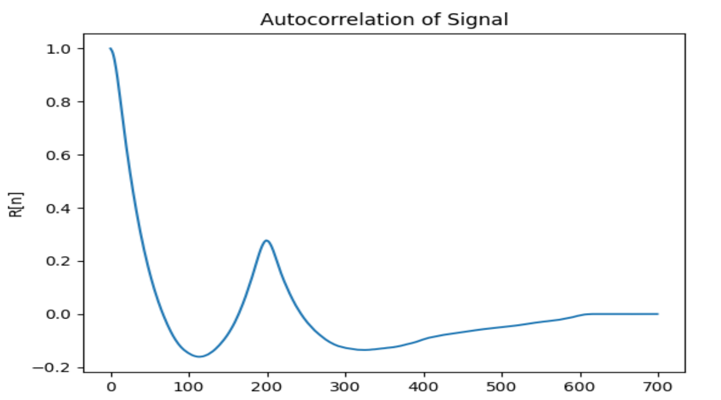
α = attenuation ratio

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

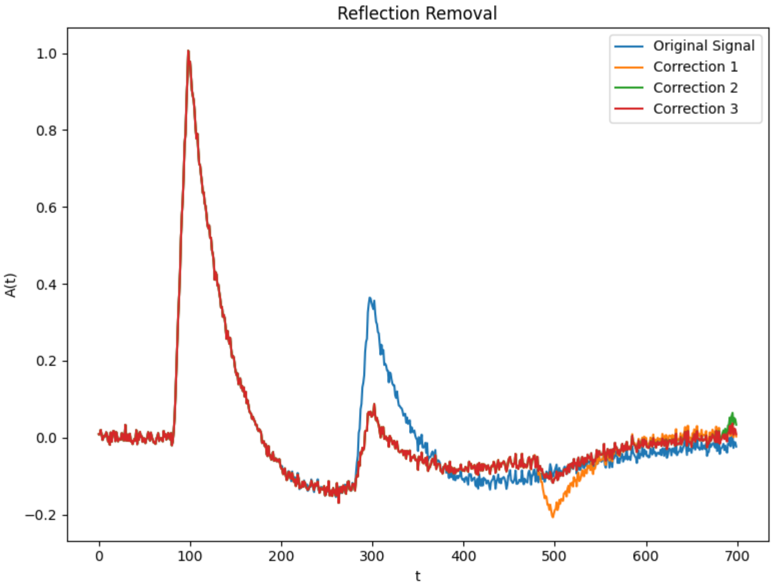
3. Delay, scale and subtract from the original signal = 1 correction

M corrections:

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



Reflection removal demo on Python (RPi)

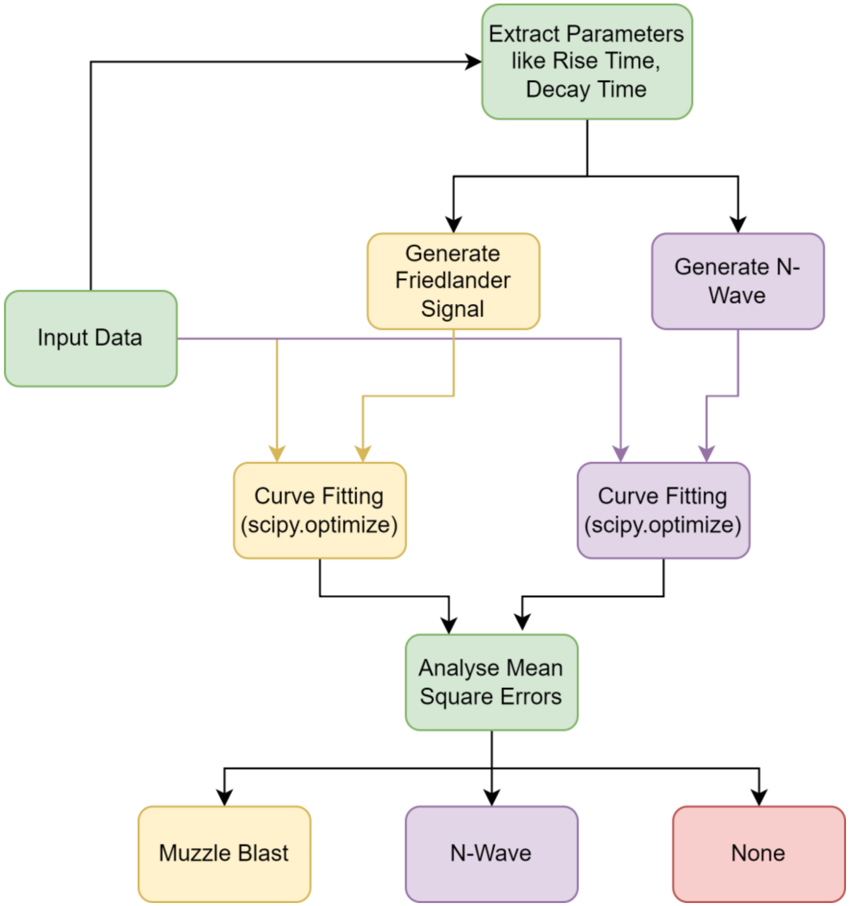


Event Categorization

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.

Demonstration of curve fitting for Muzzle Blast



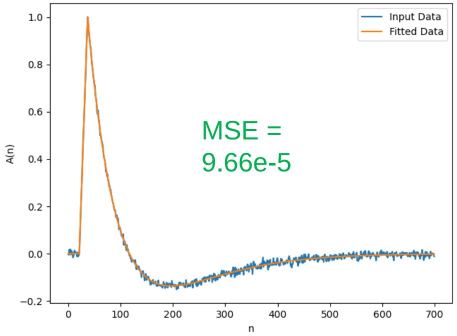
RPi Implementation of Gunshot Localization

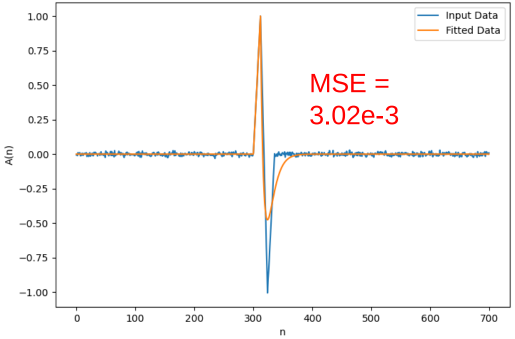
Event Categorization

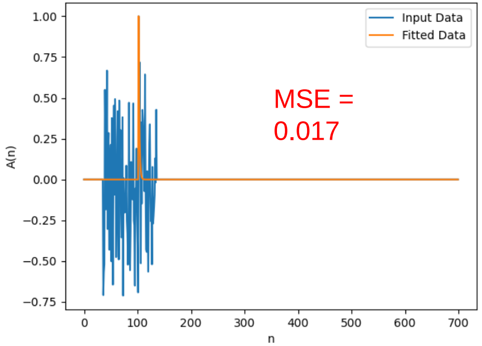
Actual Event | Muzzle Blast | Shock Wave | Noise Pulse

Fiting

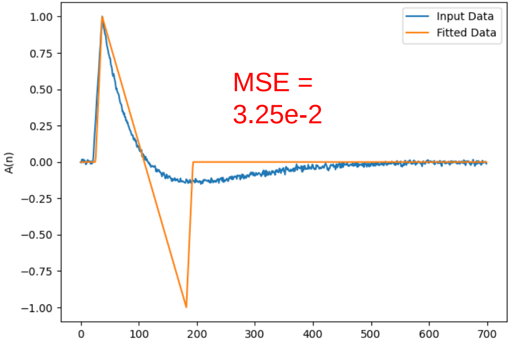
Friedlander Wave

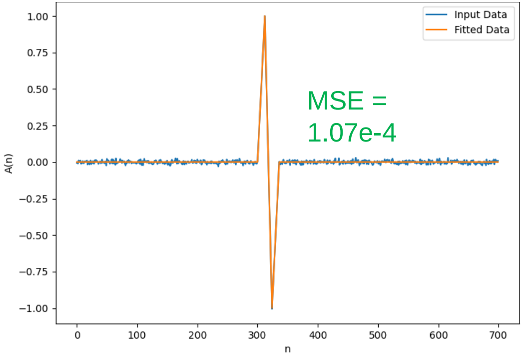


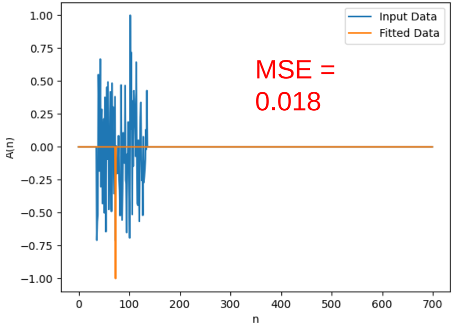




N-Wave

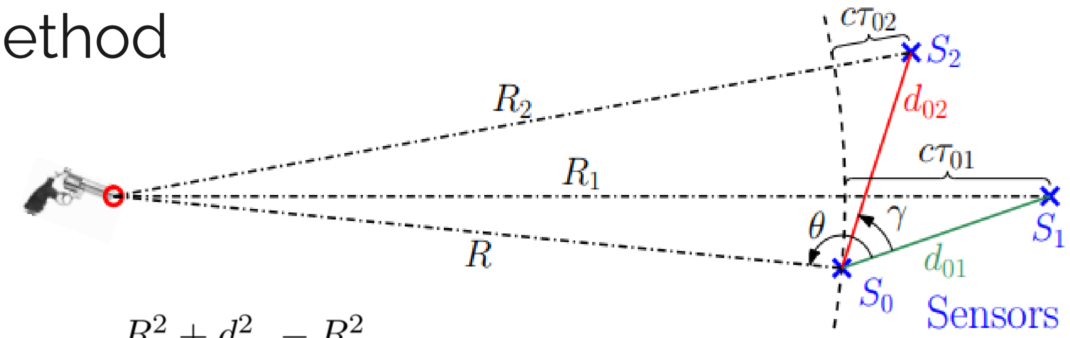






Result | Muzzle Blast | Shock Wave | None

Wavefront Curvature Method



- Planar localization technique assuming circular wavefront expansion
- Min sensors required = 3

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

$$R = \frac{d_{01}}{2}\frac{1-\eta_{01}^2}{\cos\theta + \eta_{01}} \qquad R = \frac{d_{02}}{2}\frac{1-\eta_{02}^2}{\cos(\theta-\gamma) + \eta_{02}} \qquad \eta_{ij} := c\tau_{ij}/d_{ij}$$

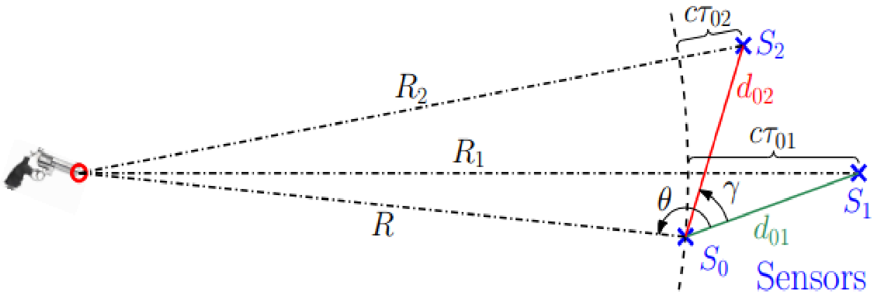
$$\theta = \tan^{-1}\left(\frac{\sin\gamma}{\cos\gamma - \delta}\right) \pm \cos^{-1}\left(\frac{\delta\eta_{01} - \eta_{02}}{\sqrt{1+\delta^2 - 2\delta\cos\gamma}}\right), \quad \delta := \frac{d_{02}\left(1-\eta_{02}^2\right)}{d_{01}\left(1-\eta_{01}^2\right)}$$

- 2 mathematically possible solutions
  Near one to be ignored
- Error:

$$\epsilon_{wc} = |R_2 - R_1 - c\tau_{12}|$$

Applying Wavefront Curvature Method

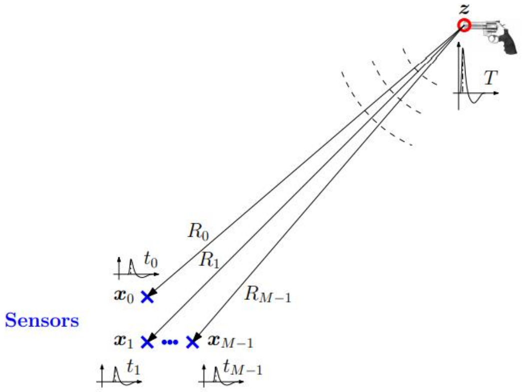
- The **WCM.py** file implements the Wavefront Curvature Method as described previously.
- This is the exact geometrical solution obtained in terms of bearing angle (θ) and range R.



- 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.

Other Algorithms for Muzzle Blast Based Localization

Non-linear least squares method



$$ct_i = D(\boldsymbol{x}_i, \boldsymbol{z}) + cT, \quad i \in \{0, 1, \ldots, M-1\}$$

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

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

c: ambient speed of sound

T: Instant of gunfire at the position z

$J_{ij}(z)$: *Jacobian*

$D(.,.)$: *Distance function*

- Several other algorithms like Non-linear least squares method, Bancroft's method and Localization using multiple direction-of-arrival vectors exist
- However, wavefront curvature method is very simple gives us the geometrically exact solution.
- In case of multiple sensors, non-linear least square method can be used for robust computations.
- We can also use this method to refine the estimate given by WCM.

Results - Error Variation with Source Location

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

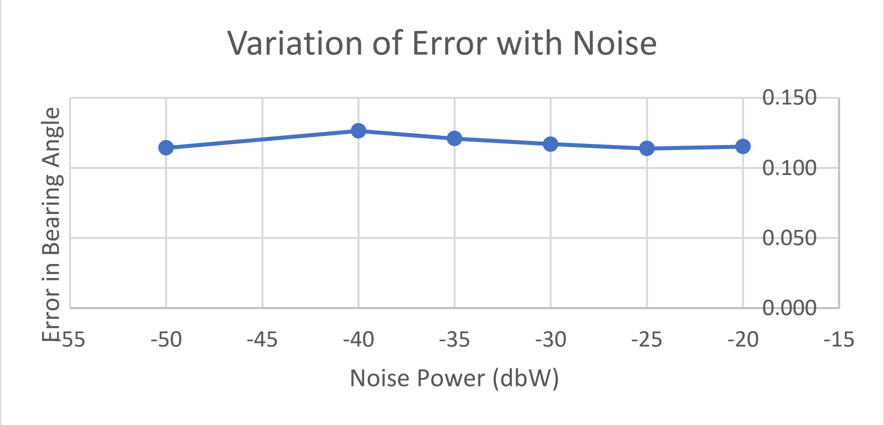
The algorithm was tried was various test cases of Source Locations (X and Y) and the error in the estimated bearing angle was computed as follows

(All Angles are in degrees)

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

Results - Error Variation with Noise Power

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



- Performance of the localization algorithm is independent of the noise to signal ratios investigated
- 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.
- For Noise Power > -15dB, the algorithm fails to classify the event as it is mostly noise.

Conclusion

- Two distinct gunfire events: Muzzle Blast & Shock Wave
- An attempt to design a simulator for gunshot and microphone array has been made in this project with features like original signatures, path losses, reflections and noise addition.
- The gunshot location can be found out using the various signal processing techniques
- Wavefront Curvature method was used after classification
- The algorithms were tested on RPi 4B.
- The bearing angle can be found out with very high precision as compared to the range of the gunfire.
- The algorithms is almost immune to noise levels and hence filtering step in the pre-processing is somewhat redundant.
- The further work in this project will be

➢ to better the estimate using non-linear least square method

➢ expand the scope for multiple sensors placed at different locations

➢ Try various other techniques including DL based approaches on real life data (fire-crackers)

References

Github Link for the Project: https://github.com/adityaharakare1/SREGunshotLoc

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*

Thank You