Generated using the official AMS LaTeX template v6.1

Comparing Active and Passive Noise Cancellation Techniques for Wind Filtering in Infrasound Detection Applications

Hakan Saplakoglu,[a] Zachary Clayton,[a] Chengzhi Wu Li,[a] Michael Zink,[a]

[a] *Paros Center for Atmospheric Research (PCAR), University of Massachusetts Amherst*

Corresponding author: Hakan Saplakoglu, hsaplakoglu@umass.edu

ABSTRACT: Infrasound signals —those below 20 Hz— are commonly found in both natural environments and human-made infrastructure. These signals can be sourced and analyzed using infrasound barometers or microphones. Often, infrasound signals carry very weak power due to their source being large distances away. Infrasound sensors can be very precise and are able to record these signals, however, in the presence of any external noise, these signals are easily lost. The most prominent source of noise in an outdoor sensing environment is wind, a non-stationary noise source. In this paper, an ultrasonic anemometer is used to measure instantaneous wind speed at the same location to where an infrasound barometer takes measurements. We demonstrate that there is a weak correlation between the anemometer noise signal and the barometer signal. Normalized least-mean-square (NLMS) adaptive filtering will be used, specifically its residual error, as the noise-cancelled output from the barometer data. It will be shown that this method is successful at reducing a portion of the wind noise in the barometer signal. However, this method is not entirely successful, as it is not able to uncover lost infrasound signals due to noise. To evaluate this, a separate barometer with a passive wind filter is deployed to serve as a comparison to an ideal condition. In addition, we show that this is a challenging problem and outline the various solutions used to overcome challenges throughout the process.

1. Introduction

In the field of infrasound measurements, clean data that has minimal noise is critical to observing these often very low power signals. Some of the signals that can be observed are: Earthquake pressure waves, wind turbines, tornadoes, microbaroms—which are infrasound from waves during storms, and HVAC equipment.

Wind is a very common and disruptive noise source for infrasound measurements, as it is significantly higher in power than most observable infrasound signals, and covers the entire infrasound frequency spectrum: 0 to 20 Hz. This results in signals that could have been observed being drowned out by wind.

There are two methods of noise cancellation: active and passive. Passive noise cancellation involves building large specialized filters. This is because passive filters must average out the turbulent wind, whose vortices can be large. The filter itself must be larger than the largest vortex, making them difficult to deploy in the field and requiring more maintenance for long-term deployments. The alternative is active noise cancellation. Active noise cancellation requires a separate sensor, such as the ultrasonic anemometer used in this paper, to record noise data near the barometer.

Active noise cancellation results in a more expensive bill of materials for an infrasound sensing deployment, but it is much easier to deploy and maintain long-term. This paper will evaluate the effectiveness of adaptive filter methods for active wind noise cancellation of infrasound barometer data.

2. Background & Related Work

a. Background

1) Infrasound Measurements

Infrasound describes very low frequency bands, specifically those below 20 Hz. In this, paper frequency bands between 0 and 10 Hz were analyzed. The work in this paper focused on capturing atmospheric infrasound signals. For example, an earthquake occurring continents away can send a pressure wave which bounces off the atmosphere and can circle the world more than twice Johnson

et al. (2020). However, a source that far away will naturally be recorded as an extremely weakly powered signal, if at all.

These infrasound signals present themselves as pressure waves, which is what is being recorded. Specifically, recording real-time static pressure at the site of an experiment is of primary interest. There are two types of pressure: static and dynamic. Dynamic pressure is influenced by the direction of a moving air mass. Dynamic pressure is not relevant for analyzing infrasound signals. The static pressure data recorded in this paper uses high-precision barometers.

2) Sensors

Both of the sensors used in this project measure similar events, but do so very differently. Barometers measure instantaneous air pressure at a given point, whereas anemometers measure the rate of displacement of an air mass in a given region. This allows for active noise cancellation to be attempted using the anemometer data as a noise signal.

This experiment uses precise DigiQuartz infrasound barometers made by ParosScientific ParosScientific (2024) to make pressure measurements. These devices use two quartz crystals: one controlled, and one exposed to ambient pressure. By running a current through these crystals and comparing their oscillation frequencies, very precise pressure measurements can be made. The DigiQuartz barometers can sample up to 40 Hz, with a range of 620 - 1100 hPa, and a precision of +/- 0.08 hPa. In this paper, the barometers are sampling at 20 Hz, which allows for frequency analysis of signals up to 10 Hz.

In order to measure wind speed to use as the recorded noise source, we use a Young Ultrasonic Anemometer Young (2024). This anemometer has no moving parts and calculates the speed at which an air mass is moving by performing time-of-flight calculations between three separate transmitters and receivers on the sensor body. The sensor having no moving parts allows the anemometer to avoid a natural smoothing of the noise data due to inertia. The Young Ultrasonic Anemometer can sample up to 20 Hz, which is the same sampling rate configured for the barometer.

3) Wind Noise

Wind can be a very powerful non-stationary noise source for infrasound measurements, especially when the sensor is deployed outdoors. This is because fluids (in this case wind) moving at different speeds have different pressures, which infrasound barometers measure. Specifically, a fluid moving

with a higher speed has lower pressure, while a fluid moving with a lower speed has higher pressure. This is Bernoulli's equation Princeton (2024). Given this relation, it is expected that the correlation between pressure data and wind speed data is strongly negative.

Wind produces especially apparent noise by introducing higher frequency signals due to the turbulent eddies formed when wind is close to the ground Bowen and Lindley (1977).

4) Adaptive Filter Implementation

Adaptive filters, specifically the least-mean-square (LMS) variant are often used for noise cancellation Dewasthale et al. (2015). In this paper, the NLMS (normalized LMS) algorithm is used. This is because NLMS is more resistant to divergence than traditional LMS. In addition, LMS adaptive filters are the most useful for non-static noise, where the noise source does not have a constant noise floor Widrow et al. (1976). Wind is a non-stationary noise source. The NLMS algorithm uses a weight update formula (equation 1) to update a linear FIR filter in an adaptive manner to achieve the desired noise cancellation.

$$W_{n+1} = W_n - \mu \nabla \varepsilon[n] \tag{1}$$

For this paper, the NLMS adaptive filter algorithm was extended in two key methods: the addition of repeated adaption per sample, and a minimum reference power check. This implementation is available as a MATLAB script on GitHub hakasapl (2024).

In the algorithm implemented for this paper, it was also necessary to allow the filter coefficient adaption to occur more than once for each sample. This allowed a more effective noise cancellation while avoiding divergence. This involved running the adaptive algorithm more than once, such that the filter coefficients are updated more than once each iteration.

b. Related Work

The related work section will briefly present some of the past work done on this topic and how it relates to this paper. There is very minimal work done specifically for infrasound active wind noise cancellation.

The work of Vazrekar presents an approach based on the cancelation of acoustic echo using adaptive filters RLS (recursive least squares) Vazrekar (2016). While this work presents the

implementation of an LMS-based (Least Mean Square) adaptive filter for noise cancelation it focuses on frequencies much higher than infrasound in the acustic frequency band.

In Dickey and Mikhael (2012) the authors present a method of isolating a seismic signal from infrasound data. This approach is the inverse of our approach, since it attempts to remove the desired signal. This only works well if the signal to be removed is known. It is our goal to create a more generic approach that can be applied for a variety of infrasound signals. Nevertheless, this work inspired our approach of active wind noise cancelation.

The work presented by Krishnamoorthy et al. (2020) demonstrated to us how passive wind filters for barometers have been applied in the past. Our paper in particular uses the same model ParosScientific ParosScientific (2024) barometers that are used in this paper. A quad-disk style passive wind filter is used, which is shown to be ineffective at canceling wind noise in this paper's experiment. As an alternative, we used a more comprehensive porous hose filter to record data for the ideal case. While this paper does not attempt active cancellation, it was important in gathering knowledge about existing solutions for passive filters.

In this masters thesis, acoustic echo cancellation using RLS (recursive least square) adaptive filters Vazrekar (2016) is presented. This publication was useful for seeing an implementation of an LMS-based (Least Mean Square) adaptive filter for noise cancellation. However, as this is in the acoustic realm, the frequencies dealt with are significantly higher than those in our current work. In addition, there is little frequency analysis and the focus is on SNR. We certainly discuss SNR in our work, however, frequency analysis is a big part of infrasound measurements so the frequency domain is more important for our approach.

3. Setup

In this section, the experimental deployment used to collect the data in this paper will be defined. This includes images of the deployment, a system-level block diagram, and technical descriptions on the various components of the system.

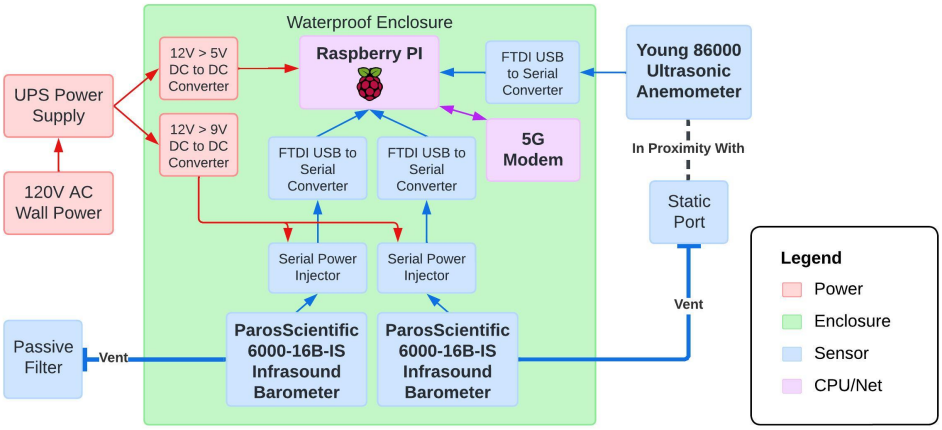


FIG. 1. System Block Diagram

a. System Overview

1) SYSTEM COMPONENTS

The sensors used in this paper were deployed long-term for data collection on a rooftop at the University of Massachusetts Amherst campus. Because of the outdoor location, the deployment includes: Raspberry PI 4, DC to DC onverters for 12, 9, and 5V rails, 2x Paroscientific 6000-16B-IS infrasound barometers, serial power injectors for infrasound barometers, 5G modem for external Internet access, Young 86000 ultrasonic anemometer,external barometer vents with static port, UPS power supply. and porous hose passive pressure filter.

An image of the deployment on the roof of a building on our campus is shown in Figure 2.

In this configuration one barometer is the experimental data collector, which has an intake port immediately next to the ultrasonic anemometer. The other barometer is connected to a porous-hose passive-wind-filter. This filter is an 8-spoke design with a 50-foot diameter. This passive filter allows collection of an ”ideal” noise cancellation output, since the passive filter is very good at removing most of the wind. An image of the passive filter is shown in Figure 3.

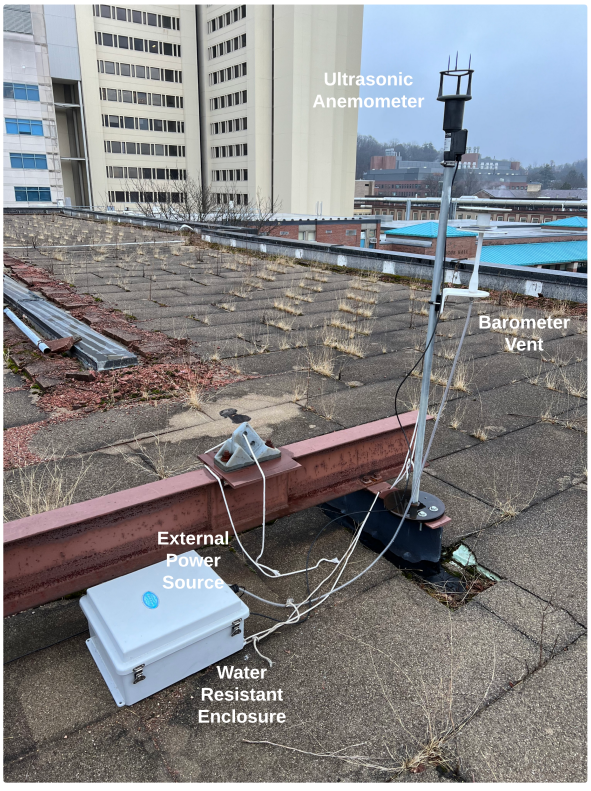


FIG. 2. Experimental Deployment on University of Massachusetts Amherst Roof



FIG. 3. Passive Wind Filter deployed for the Experiment

b. Noise Filtering Algorithm

1) TIME OFFSET CORRECTIONS

The data from all sensors must be aligned and made the same length. This allows the adaptive filter part of this algorithm to avoid worrying about alignment and vector lengths. The second goal

of this part of the algorithm is to introduce a delay in the first source passed. For this experiment, that will be the wind data. The delay is implemented to de-correlate the wind and barometer signals, which is critical for the NLMS filter to converge properly. The corrections are performed as follows:

1. Determine the number of signal sources (can be an infinite amount)
2. Create a time vector of two elements from negative infinity to positive infinity which will store the timestamp limits of all sensors
3. Determine the delay in seconds for the first signal source
4. Loop through each signal source to determine their maximum timestamp and minimum timestamp. Apply the minimum and maximum timestamps to the time vector created in (2).
5. Loop through each signal and divide each matrix into data and timestamp vectors.
   (a) Find the indices which are in between the timestamps defined in the vector in step (2)
   (b) Create a new matrix with only those indices.
   (c) Add created matrix to the output cell array

2) NLMS Adaptive Filtering

This part is the main part of the algorithm that is responsible for performing noise cancellation with an adaptive NLMS filter. This function does not perform on all the data at once. It uses a windowed method with no overlap. The window size is configurable via a parameter. In addition, all relevant tunables for the NLMS filter can be configured as parameters. This includes: filter order, step size (μ), regularization parameter (χ), number of adaptations per cycle, and minimum reference power.

A block diagram for this adaptive filter is shown in Figure 4.

This part of the algorithm has the following steps:

1. Determine the maximum length of the output, which is the number of windows multiplied by the size of the windows
2. Initialize a vector of length filter order to store filter coefficients in between loops

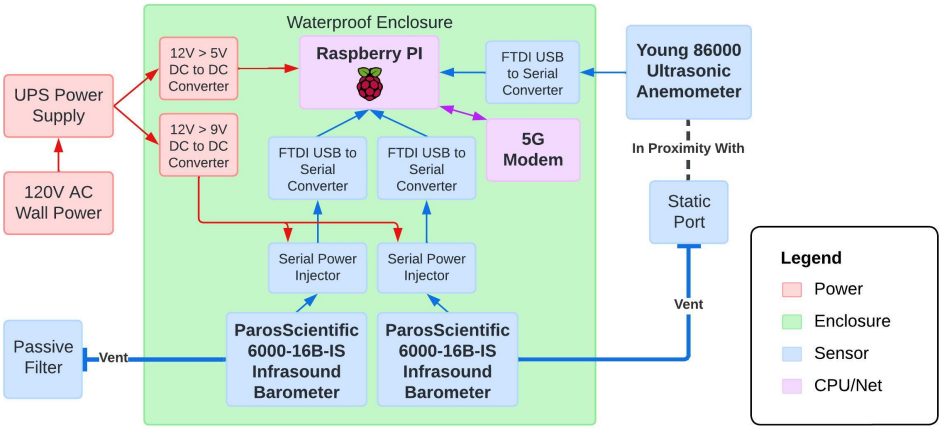


FIG. 4. NLMS Adaptive Filter block diagram

3. Start the main loop: determine the window indices and create two vectors for noise and signal based on indices

   (a) Remove the DC offset by finding the average of the entire window and subtracting it from the data vector

   (b) Call the NLMS algorithm for this window with the filter coefficients from the last loop (detailed steps for the NLMS filtering implemented in this paper is below this section)

   (c) After the NLMS is complete, save the filter coefficient values to the vector created in step (2).

   (d) Add the DC offset back to the noise-cancelled output

The NLMS algorithm steps are described as:

1. Initialize a vector to store filter coefficient weights

2. Initialize a shift register vector, which will hold the currently processing samples and slide every loop

3. Start the convergence loop: Update the shift register vector with a new sample

   (a) Start the repeated adaption loop: this loop will run as many times as the parameter repAdapt

      i. Calculate the residual error from the filter coefficients and the shift register values

      ii. Using the calculated residual error, update the filter coefficients. This is where LMS and NLMS are different. NLMS has an extra denominator derived from the value of chi, which prevents divergence.

   (b) Add filter weights to output weight vector

   (c) Add the processed sample to the output vector

This algorithm was written using MATLAB and is available on GitHub hakasapl (2024).

3) Algorithm Parameters

If the signal being analyzed is known, such as the New Jersey earthquake signal used in section 4, the algorithm parameters can be tuned to maximize the improvement for specific signals. For the known signal case, an optimization function can be developed and evaluated with existing tooling, such as the MATLAB optimization toolbox . For a known signal the optimization function will try to maximize the correlation between the output and the known signal.

For the unknown signal case, a manual process that modifies the parameters and evaluates the output is required. In the analysis performed in this paper, this was done largely qualitatively. However, performing this quantitatively with a known signal at different frequencies and powers can be accomplished by creating artificial signals, which is outlined further in section 5.

4) Window Size

The NLMS adaptive filtering applies to each window within a dataset. Each window must be sized such that data close to 0 Hz is preserved. This is because the average of the window is subtracted from each window before processing, which removes the DC offset. A smaller window results in the high-pass threshold being higher. Because many infrasound signals worth analyzing can be as low as 0.5 Hz, a window size was chosen in order to preserve these frequencies. In this paper 512 samples was the chosen window size, which is approximately 25 seconds at the 20 Hz sampling rate.

5) Filter Order

Increasing the filter order results in a slower runtime, but a more precise output. Increasing this value beyond a certain threshold results in no noticeable improvement and solely increases runtime of the algorithm. For the dataset used in this evaluation A filter order of 64 was chosen for this paper.

6) Step Size

Step size adjusts the rate of gradient descent between each iteration. Increasing this value eventually can lead to divergence. It is important to note that divergence may occur locally and re-converge later, which is often undesirable as well. A step size was maximized by evaluating the change in filter coefficients over time to visualize any divergence. A step size of 0.4 was chosen for this paper.

7) Regularization Parameter

The regularization parameter is used to add a cost to each gradient descent step. This parameter is useful to prevent divergence. The minimum value where divergence does not occur should be used. In this case, the value chosen was $1 * 10^{-7}$.

8) Repeated Adaption

Repeated adaption is the number of times the filter is adapted for each sample. Increasing this value decreases the power of the output. After a certain threshold the power of the whole output is decreased rather than the power of just the noise trying to be removed. A balanced value of 3 was used for this paper.

9) Minimum Reference Power

The minimum reference power should be set to the power threshold where noise is considered to exist. This was determined by finding the power of a no-noise area as well as a noise area. A value of $1 * 10^{-3}$ was used.

TABLE 1. Algoritm parameters

Parameter	Value
Window Size	512
Filter Order	64
Repeated Adaption	3
Step Size (μ)	0.4
Chi (χ)	$1 * 10^{-7}$
Minimum Reference Power	$1 * 10^{-3}$

4. Evaluation

In this section, the performance of the noise cancellation algorithm introduced in Sect. 3 is evaluated. Several data sets that were collected from the setup described in Sect. 3 are used for this evaluation.

At the location where these sensors are deployed, there is often a very weak but visible infrasound signal just above 7 Hz, which is attributed to the HVAC systems nearby. Throughout this analysis we will be observing this signal and verifying that it is not being removed by the noise-filtering algorithm.[1]

The following noise-cancellation algorithm parameters shown in Tabel 1 were used, which were found to be the most ideal for these data.

a. Qualitative Analysis

In this section, time and frequency domain plots will be compared qualitatively before and after noise cancellation. This section will show data from a 24 hour period from March of 2024.

1) TIME SERIES COMPARISON

Figure 5 shows the time series comparison between input and output. It is clear that the output has much less power than the input but still follows the general trend. It appears to be much closer to the passive filter data. Looking at a zoomed in area of the time-series plot in Figure 6, it is clear that much of the turbulence from the input pressure signal has been canceled and the output looks very similar to the passive filter data, only with a differing DC offset, which is the current atmospheric pressure in this case.

[1] While it might be seen as noise, we treat this signal as an infrasound source for this evaluation.

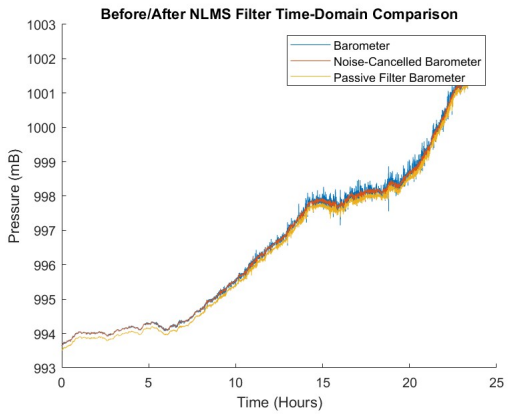


FIG. 5. 2024-04-06 Time-Series Comparison

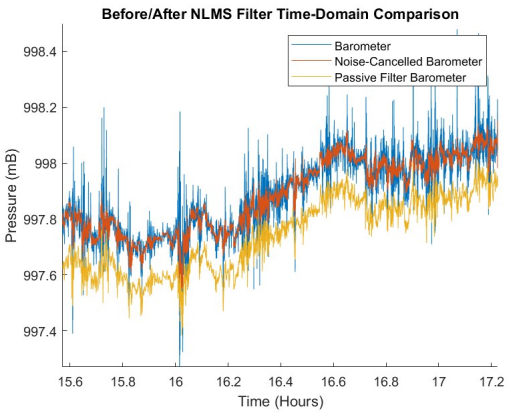


FIG. 6. 2024-04-06 Time-Series Comparison Close-up

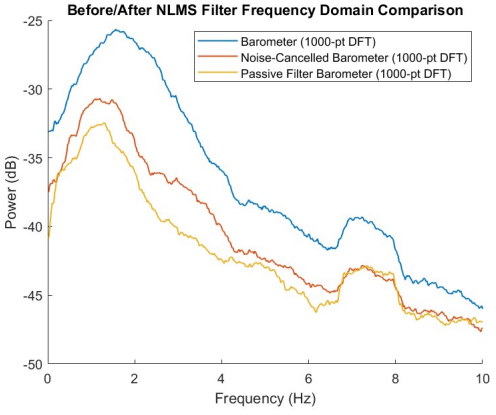


FIG. 7. 2024-04-06 DFT Plot

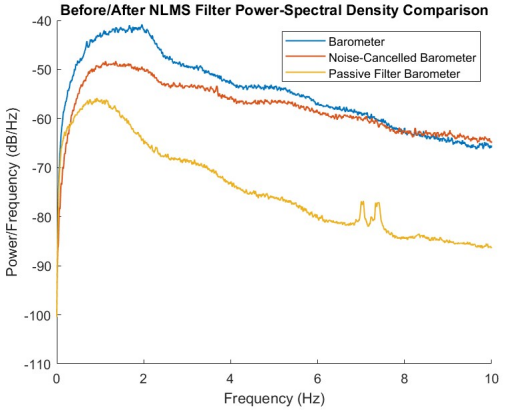


FIG. 8. 2024-04-06 PSD Plot

2) FREQUENCY DOMAIN COMPARISON

In this section, the discrete-time Fourier transform and the power spectral density will be compared. Lastly, the spectrograms will be compared to show any time-varying signals in the frequency domain.

Looking at Figure 7, it is clear that the frequency domain profile of the noise-canceled signal is much closer to the ideal condition—the passively filtered barometer—than to the input. This shows that the noise-cancellation algorithm is adjusting the power properly, especially in the frequencies 6 Hz and higher, where the DFT profile matches nearly perfectly.

In the power-spectral density plot shown in Figure 8, the passive filter barometer signal shows very apparent spikes around 7 Hz. The spectrogram in Figure 11 shows these spikes as well. The

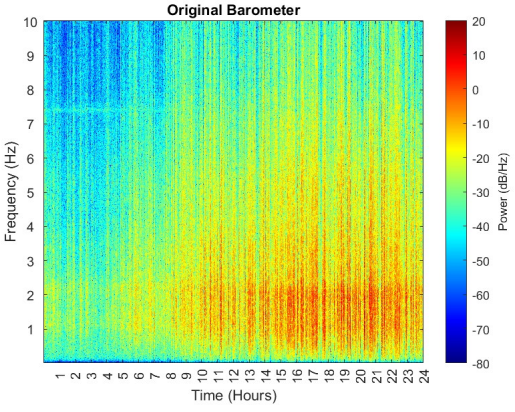


FIG. 9. 2024-04-06 Spectrogram Before Noise-Cancellation

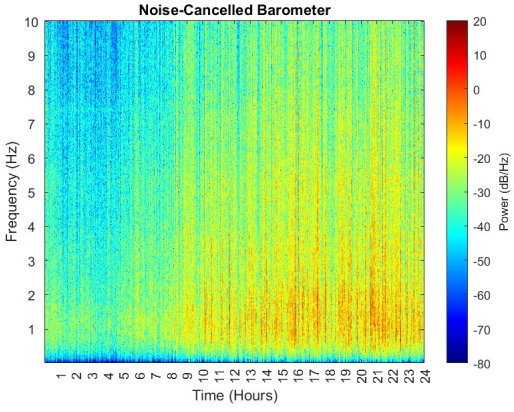


FIG. 10. 2024-04-06 Spectrogram After Noise-Cancellation

hope was that the noise-cancellation algorithm could expose these spikes in the output signal. This, unfortunately, does not appear to be the case, which is likely due to the higher frequency wind noise being harder to correlate with the input. The poor performance for these signals is attributed to the random nature of turbulent wind and how it manifests differently in two different types of sensors.

In figures 9 and 10 it becomes clear that the power of the high-wind areas is being decreased, especially in the lower frequency ranges up to 4 Hz. This results in the wind noise having more consistent power across the spectrum, rather than most of the power residing in the frequency ranges below 4 Hz. It is also important to note that the HVAC signal around 7 Hz is not being removed in the output signal.

Figure 11 shows a spectrogram covering only the frequency ranges between 6.5 and 7.5 Hz. The HVAC signal is shown very clearly, which shifts by about 0.5 Hz partway through. These two frequencies are the reason spikes are observed on the power-spectral density plot in Figure 8 for the passively filtered data. Next, this same zoomed-in spectrogram will be visualized for both the input and output signals from the noise-cancellation to determine if this signal can be recovered.

From the comparison of figures 12 and 13, it is clear that this signal is not being recovered, which is the same conclusion when looking at the power-spectral density in Figure 8. This result is unfortunate, and it confirms that, especially at higher frequency ranges, this method and sensor

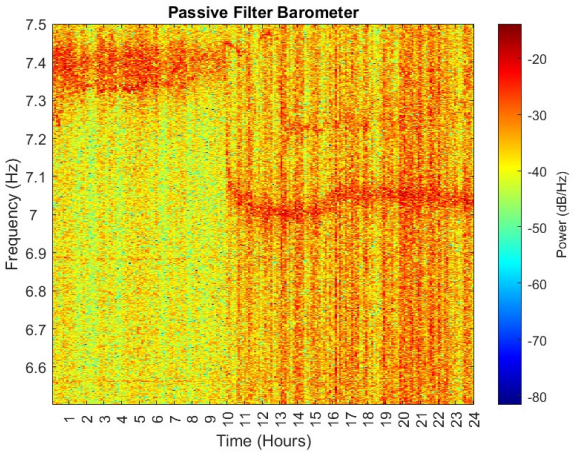


FIG. 11. 6.5 - 7.5 Hz Zoomed Spectrogram for Passive-Filtered Barometer

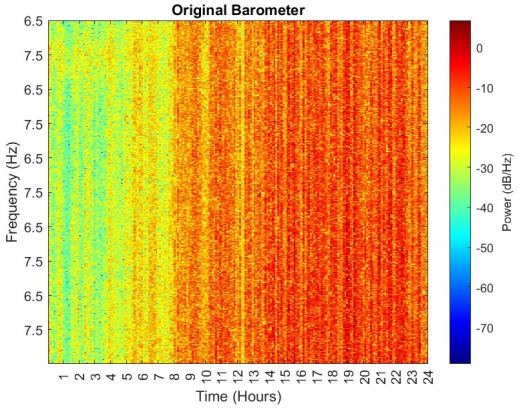


FIG. 12. 2024-04-06 Spectrogram Zoomed Before Noise-Cancellation

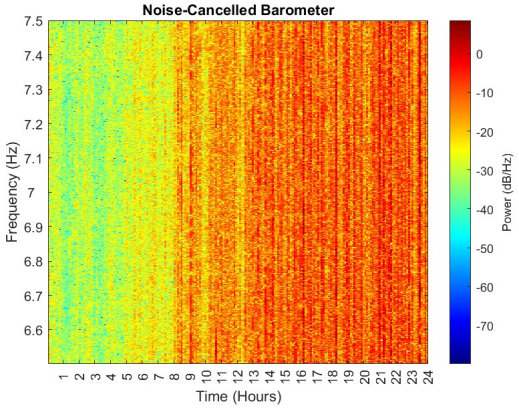


FIG. 13. 2024-04-06 Spectrogram Zoomed After Noise-Cancellation

combination are not suitable to extract unknown signals. The next section will show the results when trying to extract a known signal.

b. Segmental SNR Results

In this section, the segmental SNR results will be compared for each dataset before and after noise cancellation. The segmental SNR will compare the input barometer signal with the passive-filtered barometer signal, as well as the output barometer signal with the passive-filtered barometer signal. It is expected that the output signal will be closer to the passive-filtered signal. Therefore, an increase in the segmental SNR value is expected.

TABLE 2. Segmental SNR Results

Dataset	segSNR Before	segSNR After	segSNR Diff
2024-04-01	41.5855	46.1305	4.5450
2024-04-06	33.8363	43.1959	9.3596
nj-earthquake	9.5077	19.5914	10.0836

TABLE 3. Correlating Output to Input

Dataset	Wind and Baro	Output and Wind	Output and Baro
2023-03-09	0.5248	0.3518	0.4276
2024-03-29	0.4632	0.3419	0.4210
2024-04-01	0.4569	0.3679	0.4590
2024-04-06	0.4477	0.3273	0.4096
nj-earthquake	0.4322	0.3023	0.4435

Table 2 shows that for every relevant dataset, there is an increase in segmental SNR. This shows that the output of the noise cancellation algorithm is closer to the ideal-conditioned passive-filtered barometer when compared to the raw input data.

1) Comparing Output to Input

In this section, the output pressure signal will be correlated to both inputs (pressure and wind) in an effort to show that the correlation with the original wind data decreases and the correlation with the barometer data remains weakly correlated.

Evaluating the results in Table 3 (all values are normalized to 1), for every dataset the output's correlation with the input noise (wind), is decreased. As expected, these results are even more pronounced when correlating the entire dataset with each other instead of using a windowing method, however, this paper used the windowing method for other tables, therefore it would not be a useful comparison to correlate the entire dataset.

c. Noise Cancellation with a Random Noise Source

It is important to verify that the noise cancellation is functioning correctly, and not simply reducing the power of the original signal. To verify this, we use a vector of random numbers of the same length as the anemometer input vector, and analyze the output. Running the NLMS algorithm with the same parameters, the filter still converges, but the filter coefficients converge to 0, which is consistent with a random input.

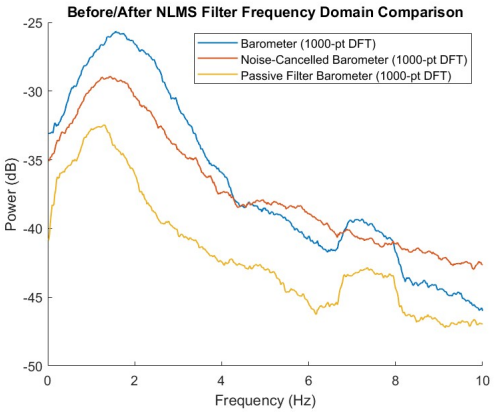


FIG. 14. Random Data DFT Comparison

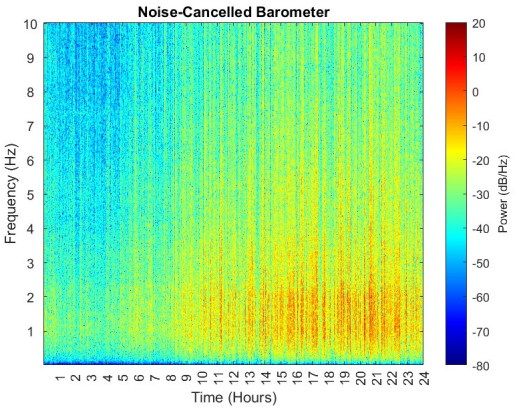


FIG. 15. Random Data Noise-Cancelled Spectrogram

Looking at Figure 14, it is clear that the output's frequency domain is not similar to the ideal, which is expected. Also, the spectrogram in Figure 15 shows that the HVAC signal at 7 Hz is almost completely missing in the output, which means the algorithm just decreased the power of all frequencies, not just the wind, which is also expected. This shows that the algorithm is accomplishing some wind noise-cancellation when correct data is used.

5. Conclusion & Future Work

a. Conclusion

In this paper, an experimental setup involving infrasound pressure measurements was presented with the hardware required to attempt active noise cancellation on infrasound barometer data. A modified normalized LMS adaptive filter algorithm was implemented, which is able to partially reduce wind noise in barometer data. In both frequency and time analysis of the residual error output, it was found that wind is being canceled, however, often times not being canceled sufficiently to allow for hidden data to be revealed. This is primarily due to the noise data, in this case the anemometer signal, not being correlated enough with the wind noise portion of the barometer data. This is very likely due to the turbulent nature of wind, especially when in close proximity to the ground. Lastly, this paper shows that active noise cancellation of infrasound signals is a hard problem, one that has yet to have an ideal solution.

b. Future Work

There are several challenges we would like to improve moving forward. These include:

- Continue to improve performance of the noise-cancellation, especially in the lower frequency bands, where the power of the wind noise is most apparent.
- Evaluate different experimental deployments. For example, the sensors being higher up on a pole could reduce some of the more turbulent effects of wind close to the ground.
- Attempt to use the algorithm developed in this paper with different kinds of sensors.
- Develop a process for creating reproducible pressure waves as a control signal for validation purposes.

Acknowledgments. Funding for this work was provided by Jerome M. and Linda L. Paros whose generosity established the Paros Center for Atmospheric Research.

References

Bowen, A. J., and D. Lindley, 1977: A wind-tunnel investigation of the wind speed and turbulence characteristics close to the ground over various escarpment shapes. *Boundary-Layer Meteorology*, **12 (3)**, 259–271, https://doi.org/10.1007/BF00121466.

Dewasthale, M. M., R. D. Kharadkar, and M. Bari, 2015: Comparative performance analysis and hardware implementation of adaptive filter algorithms for acoustic noise cancellation. *2015 International Conference on Information Processing (ICIP)*, 124–129, https://doi.org/10.1109/INFOP.2015.7489363.

Dickey, J. T., and W. B. Mikhael, 2012: An adaptive technique for isolating the seismic response of an infrasound sensor. *2012 IEEE 55th International Midwest Symposium on Circuits and Systems (MWSCAS)*, 1028–1031, https://doi.org/10.1109/MWSCAS.2012.6292198.

hakasapl, 2024: 2024-ms-thesis. GitHub, https://github.com/hakasapl/2024-ms-thesis.

Johnson, J. B., T. D. Mikesell, J. F. Anderson, and L. M. Liberty, 2020: Mapping the sources of proximal earthquake infrasound. *Geophysical Research Letters*,

47 (23), e2020GL091 421, https://doi.org/https://doi.org/10.1029/2020GL091421, https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2020GL091421.

Krishnamoorthy, S., D. C. Bowman, A. Komjathy, M. T. Pauken, and J. A. Cutts, 2020: Origin and mitigation of wind noise on balloon-borne infrasound microbarometers. *The Journal of the Acoustical Society of America*, **148 (4)**, 2361–2370, https://doi.org/10.1121/10.0002356, https://pubs.aip.org/asa/jasa/article-pdf/148/4/2361/15346564/2361_1_online.pdf.

ParosScientific, 2024: Paroscientific, Inc. URL https://paroscientific.com/.

Princeton, 2024: Bernoulli's Equation. URL https://www.princeton.edu/~asmits/Bicycle_web/Bernoulli.html.

Vazrekar, A. V., 2016: Adaptive filter design for wind noise cancellation with tms320c6713 digital signal processing kit. Ph.D. thesis.

Widrow, B., J. McCool, M. Larimore, and C. Johnson, 1976: Stationary and nonstationary learning characteristics of the lms adaptive filter. *Proceedings of the IEEE*, **64 (8)**, 1151–1162, https://doi.org/10.1109/PROC.1976.10286.

Young, 2024: Ultrasonic Anemometer | R. M. Young Company. URL https://www.youngusa.com/product/ultrasonic-anemometer-4/.