Painting with Music

Weijian Zhou
University of Toronto
Toronto, Canada

Parham Aarabi
University of Toronto
Toronto, Canada

***Abstract*—In this paper, we proposed and built an inexpensive, portable yet reasonably accurate local area sound localization system with two microphone arrays. We evaluated different array achitectures and demonstrated that a two array system outperforms a single array system of similar size. The proposed system localizes in 80 millisecond and achieved an average error of less than 3 cm for both point localization and movement tracking in a local one meter by one meter region.**

I. Introduction

Accuracte indoor locazation allows creation of noval applications with surrounding awareness that uses position and movment information as input. One application is to allow users to draw with music without physically touching the computer. Another example is to build AI games with physical pieces such as toy car racing where the computer controls some them. In this work, we aim to build such a source localization system that is portable, inexpensive, yet resonably accuracte for localization in a small area.

Global Positioning System (GPS) is the prevailing technology used for outdoor localization. Commercial grade GPS has an average error of a few meters, depending on the size and quality of the receiver [1]. While accuracy in this range is good for many applications including driving navigation and vehicle tracking, it does not provide enough precision for local movement tracking. Ultrasound based indoor localization approaches on the other hand, has achieved sub-centimeter accuracy [4]. However, ultrasound systems require the use of expensive transducers.

Bluetooth and Wi-Fi based technologies have gained popularity in indoor positioning recently, mainly due to the widespread deployment of bluetooth tags and Wi-Fi stations in public spaces. In these systems, signal strength received from different base stations are used for the estimation of the device location. However, their reported accuracy are in the range of 1 to 5 meters [2], [3], which is not enough for local movement tracking.

In this project, we have built a localization system with reasonable high precision for small area using microphone arrays that localize typical audio source. Our system is built with inexpensive electret microphones mounted on portable frames. Users can interact with our system using any device that has audio output such as a mobile phone.

In this paper, we first discussed in Section II some prior relevant research and approaches in sound localization. In Section III, we evaluated different array achitectures and their impact on localization accuracy. We demonstrated that a two array system outperforms a single array system of similar physical dimension. In Section IV, we presented the chosen architecture along with hardware details. Finally, experiment details and results are presented in Section V.

II. Background

Acoustic localization has been researched extensively in the literature. Localization techniques can be broadly categorized into Location Template Matching (LTM) and Time Difference of Arrival (TDOA) based approaches.

A. LTM

In LTM based approaches, acoustic templates acquired from different locations are first stored in the system during a "training" phase. Localization can be performed by comparing the incoming waveform with the stored templates, and the location with the best matching template is chosen as the output. Different ways of extracting templates from raw acoustic source and different similarity measures have been investigated in the past.

[5] and [6] investigated using max value from cross-correlation as a similarity measure to localize user tap on interactive surfaces. [7] used L2 distance in the Linear Predictive Coding coefficient space as a similarity measure to localize taps on surfaces. [8] further explored accuracy improvement by using multiple templates for each location and speed improvement by merging multiple templates into one representative template.

The requirement of having a template for each location to be detected makes this approach too restrictive for our project, since we want the localization to be continuous in a 2D region. Moreover, the need to recalibrate all locations during setup is too cumbersome for the end users in a portable system. Therefore, our main focus will be on TDOA based approaches.

B. TDOA

TDOA approaches exploit the difference of arrival time between the acoustic source and two fixed microphones on the plane. It can be easily shown that the acoustic sources with the same TDOA to two fixed microphones on the plane form a hyperbola. When you have more than two microphones, each pair would give a different hyperbola. The intersection of all the hyperbolas marks the source location. TDOA approaches rely on accurate estimates of arrival time differences between microphones.

In [9], authors used eight microphones mounted on the corners of a ping pong table to localize points where the ball hits the table. They used a threshold to determine the arrival time of acoustic signal. This approach works well in noise free

environment but the performance degrades with background noise. Their approach also suffers from dispersive deflections that arrive before the main wavefront of the acoustic signal. To make it more robust, authors in [13] and [14] extracted descriptive parameters for each significant peak(e.g., peak height, width, mean arrival time). The algorithm then used extracted parameters to predict arrival time with a second order polynomial, the parameters of which were fitted during calibration at fixed locations.

Cross-correlation has also been used to measure signal arrival time differences[10], [11], [12]. Cross-correlation with prefilterings is known as *generalized cross correlation (GCC)*. Different prefilterings have been investigated to improve arrival time difference estimation [15], [16], [17].

Under the GCC framework, the arrival time difference t_0 between two signals $x_1(t)$ and $x_2(t)$ can be estimated as:

$$t_0 = \arg\max_{\tau} R_{x_1x_2}(\tau) \quad (1)$$

$$R_{x_1x_2}(\tau) = \int_{-\infty}^{\infty} W(\omega)X_1(\omega)X_2^*(\omega)e^{j\omega\tau}d\omega \quad (2)$$

, where $X_1(\omega)$ and $X_2(\omega)$ are Fourier Transform of $x_1(t)$ and $x_2(t)$. $W(\omega)$ provides a way to prefilter signals passed to the cross correlation estimator. We focused on three ways of prefiltering the signal:

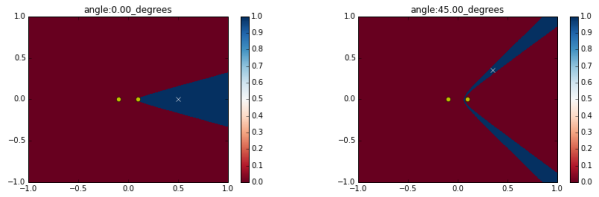
GCC	$W(\omega) = 1$. No prefiltering is done. This is unfiltered normal cross correlation.
GCC_PHAT	$W(\omega) = \frac{1}{\lvert X_1(\omega)X_2^*(\omega)\rvert}$. Each frequency is divideded by its magnitude. Only phase information contributes to delay estimation.
GCC_PHAT_SQRT	$W(\omega) = \frac{1}{\lvert X_1(\omega)X_2^*(\omega)\rvert^{0.5}}$. This is somewhere between GCC and GCC_PHAT. part of magnitude information is included in delay estimation.

III. Array Architecture

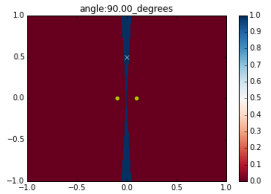
As was mentioned in the previous section, points with the same TDOA to two fixed locations form a hyperbola on a 2D plane. However, in practical systems we can only measure TDOA up to a precision. Therefore we look at all points with difference of distance close to some target value within measurement error ϵ. This ϵ represents accuracy on the measurement of difference of distances, and in practice it is related to sampling rate and estimation of difference of arrival time. In this section we evaluate the impact of difference of distance estimation on localization accuracy.

To see how precision affects localization accuracy, we simulated two microphones placed at: $M_1 : (x = -10 \text{ cm}, y = 0 \text{ cm})$ and $M_2 : (x = 10 \text{ cm}, y = 0 \text{ cm})$. A test sound source is emitted at point P which is 50 centimeters away from the origin $(0, 0)$. Fig 1 shows the region R where all points $\hat{P}$ satisfy:

$$R : \{\hat{P} : |(\hat{P}M_1 - \hat{P}M_2) - a| < 1 \text{ cm}\}$$



(a) source at ($r = 50$ cm,$\theta = 0$ degrees) (b) source at ($r = 50$ cm,$\theta = 45$ degrees)



(c) source at ($r = 50$ cm,$\theta = 90$ degrees)

Fig. 1: Uncertainty region

, where a is the difference of distance between PM_1 and PM_2:

$$PM_1 - PM_2 = a$$

Intuitively, points in R have difference of distance very similar to each other. Looking at fig 1, we can still see that R has the shape of a hyperbola, but with an uncertainty region around it. The thickness of the uncertainty region is not uniform around the hyperbola, the farther away the point is, the larger the uncertainty region becomes. This indicates for the the same delta distance movement it will generate smaller difference of distance change when the source is farther away from the array. The size of the uncertianty region is also angle dependent: points closer to the line connecting microphones have larger region compared to points close to the line bisecting microphones.

This can also be seen analytically. Assuming two microphones are placed on the x-axis at $M_1 : (-c, 0)$ and $M2 : (c, 0)$. All points $P : (x, y)$ with difference of distance $|PM_1 - PM_2| = 2a$ satisfies:

$$\frac{x^2}{a^2} - \frac{y^2}{c^2 - a^2} = 1 \quad (3)$$

To see how difference of distance changes with respect to distance, we can expand the equation and find partial differential $\frac{\partial a}{\partial x}$:

$$\frac{\partial a}{\partial x} = \frac{x(c^2 - a^2)}{a(x^2 + y^2 + c^2) - 2a^3} \quad (4)$$

Since all points in equation 4 must lie on the hyperbola, we can substitute 3 into 4:

$$\frac{\partial a}{\partial x} = \frac{c^2 - a^2}{\frac{c^2}{a}x - \frac{a^3}{x}} \quad (5)$$

The denominator of equation 5 increases monotonically as $|x|$ increases, which indicates $\frac{\partial a}{\partial x}$ decreases as we move farther

away along the hyperbola. The same distance move δx would generate smaller change in difference of distance a when the source is farther away from the microphones.

With more than two microphones, each pair of microphones generates a hyperbolic region and localization becomes finding the intersection of hyperbolic regions. The smaller the intersection region, the better the localization accuracy. To see how accuracy changes with array placement and sound source location, three microphones are placed at three vertices of an 20 cm equilateral triangle. An audio source is placed at 20 cm away from the center of the array. Fig 2 shows the intersection of regions for 5 different placement of the sound source. It can be seen that accuracy is worse when sound source is close to the line connecting any two microphones. This observation is consistent with two microphone case, since points close to lines connecting microphones have a larger uncertainty region.

To see how sound source distance affects localization accuracy, the same simulation is carried out with the sound source moved from 20 cm to 80 cm away from the center of the array. Results are presented in fig 3. Comparing with fig 2, accuracy decreases as the distance to the array increases. This is also consistent with our observation in 2 microphone case where source farther away would result in larger uncertainty region.

Intersection area is a measure of the localization accuracy. To evaluate an array's accuracy in a region, we can place sound source at predetermined grid points in the region and look at the intersection area for each tested point in the grid. The center location of intersection region can be used as localization estimate to calculate localization error. Results for a few different microphone array configurations are presented in fig 4.

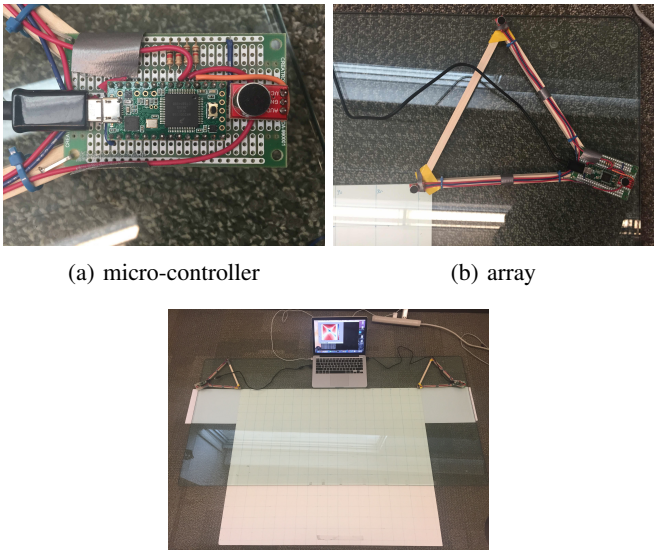
Fig 4a shows the accuracy when microphones are placed at three vertices of a 20 cm equilateral triangle. The region inside the array has good accuracy. However, for regions along line of any two microphones, the accuracy drops significantly. Average error across the region is 18.6 cm.

To evaluate how adding one microphone(without increasing array size) improves accuracy, another microphone is added to the array at $(0, 0)$. Result is presented in fig 4b. Addition of the new microphone only slightly improved the accuracy around the array region. Average error dropped from 18.6 cm to 17.1 cm. Regions near lines of microphones still have significantly larger uncertainty region.

To evaluate array size's impact on accuracy, the size of original array from fig 4a is increased by a factor of 2. The result is presented in fig 4c. The overall uncertainty area decreased across the region. Average error improved to 10.04 cm.

In fig 4d, three microphones are placed 10 cm apart from each other on x-axis. Error heatmap shows high uncertainty on the x axis, and the overall accuracy is not as good as that with three microphones placed in a triangle. The average error is 55.05 cm.

To further increase the distance between microphones, we placed four microphones at four corners of the region. Fig 4e shows the result. With this configuration, accuracy is consistently good across the region. The average error is 0.05 cm. However, placing microphones far apart at corners of the region requires accurate placement of all four individual microphones. The system is less portable compared to small arrays with microphones near each other. Placing microphones far apart from each other also causes problems in TDOA estimation, because sampling of microphones in the same array requires synchronized clock.



(a) micro-controller (b) array

(c) two arrays

Fig. 5: Localization arrays

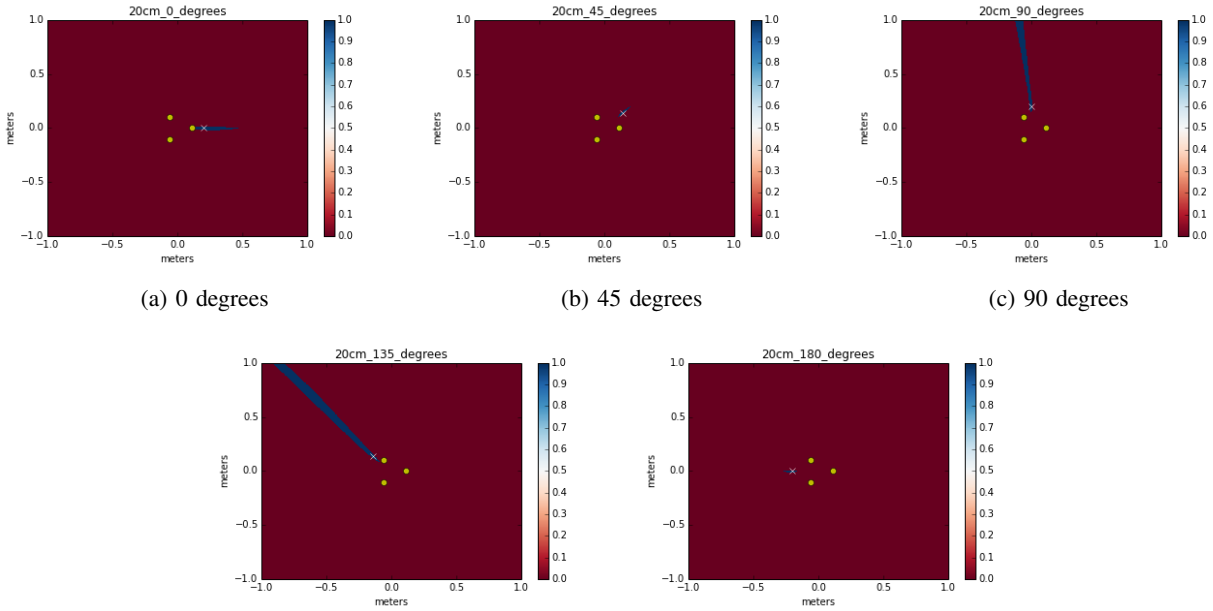
To avoid the need to accurately place four microphones at far distances(as required by fig 4e), we explored configuration with two arrays. Two 3 microphone array are placed 1 meter apart and the result is presented in fig 4f. The result indicates that this configuration has good accuracy when source is close to the arrays. Accuracy decreases as sound source moves outside of the one meter by one meter region. The average error is 2.60 cm.

With simulation results, we decided to build the two array system as described in fig 4f. The setup is reasonably portable (compared to fig 4e), while at the same time having significantly better accuracy compared to one array systems.

IV. System Setup

The end system has two arrays, each with three microphones mounted on vertices of a 20 cm equilateral triangle. A micro-controller is also attached to one of the vertices. Fig 5 shows a picture of the array setup. Micro-controller used in this project is *teensy 3.1*. It has 64k RAM memory and the ADC is capable of sampling at 500kHz. In this project, the micro-controller collects microphone data on all three channels for 12 millisecond and then send the recorded data to a computer through USB port for localization.

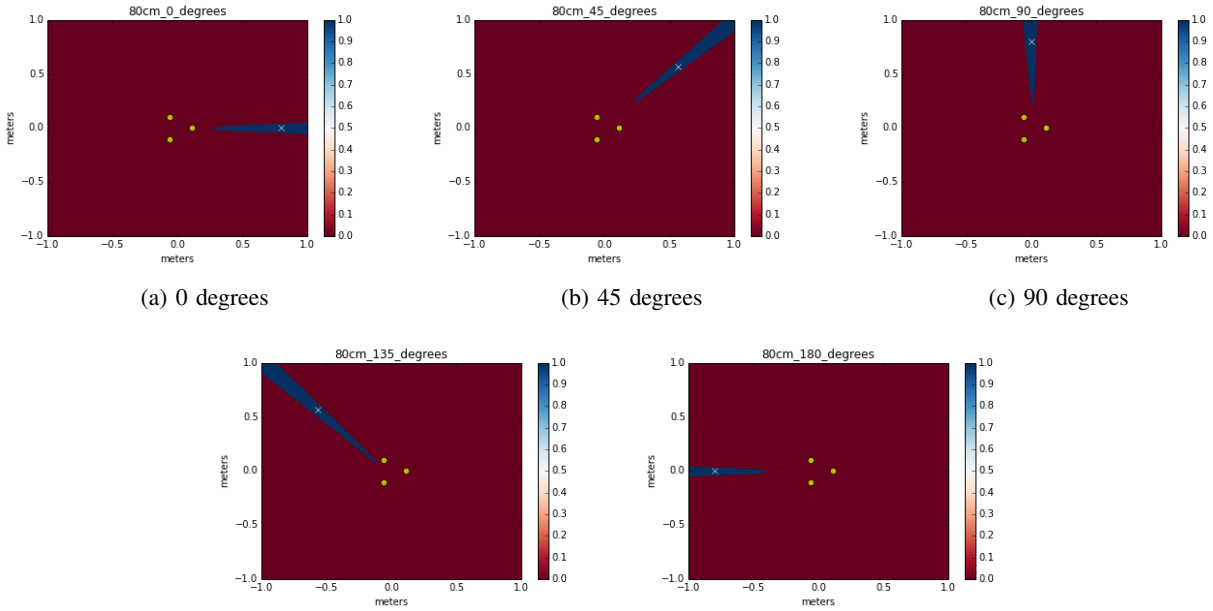
To handle uncertainty in TDOA estimation, instead of using point estimate that maximizes equation 1, we take the cross-correlation function 2 as a mapping from delay to liklihood. For each microphone array, we build a heatmap for the region.



(a) 0 degrees (b) 45 degrees (c) 90 degrees

(d) 135 degrees (e) 180 degrees

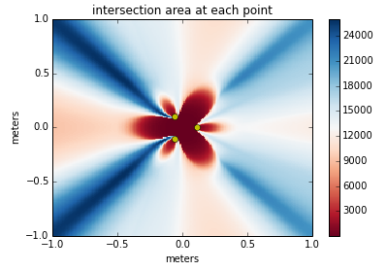
Fig. 2: 20cm equilateral triangle array. Source is 20cm away from the array



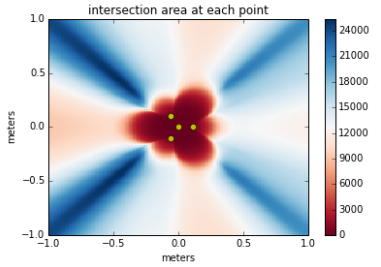
(a) 0 degrees (b) 45 degrees (c) 90 degrees

(d) 135 degrees (e) 180 degrees

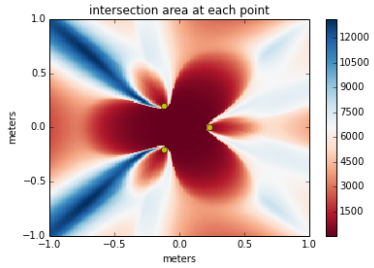
Fig. 3: 20 cm equilateral triangle array. Source is 80 cm away from the array



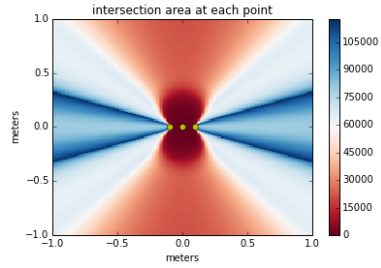
(a) 3 microphones placed at vertices of an 20cm equilateral triangle. Average error is 18.6 cm



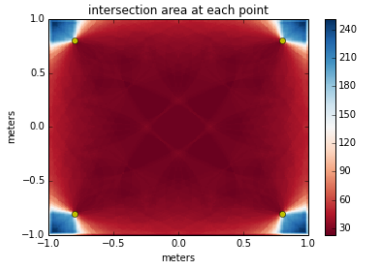
(b) another microphone added at origin. Average error is 17.1 cm



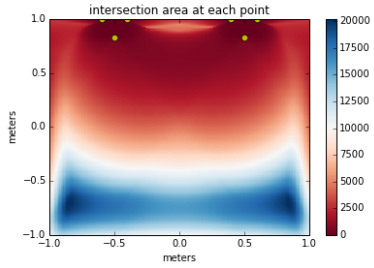
(c) 3 microphones placed at vertices of an 40cm equilateral triangle. Average error is 10.04 cm



(d) 3 microphones placed in a line. Total length is 20 cm. Average error is 55.05 cm



(e) 4 microphones placed at 4 corners of the region. Average error is 0.05 cm



(f) two 3 microphone arrays placed 1 meter apart. Average error is 2.60 cm

Fig. 4: Accuracy for different array configurations

The intensity at each point represents the likelihood of it being the source. For each point on the grid, the theoretical TDOA to each microphone pair can be precomputed. Then the heatmap can be generated by going through points on the grid and perform a lookup using equation 2. With three microphones m_1, m_2, m_3, there are three microphones pairs: m_1m_2, m_1m_3, m_2m_3. Theoretical TDOA from each location (x, y) to each microphone pair is precomputed and stored in $D_{m_1,m_2}(x, y)$, $D_{m_1,m_3}(x, y)$, and $D_{m_2,m_3}(x, y)$. Then the liklihood map $L(x, y)$ can be built as:

$$L(x,y) = R_{m_1,m_2}(D_{m_1,m_2}(x,y)) + R_{m_1,m_3}(D_{m_1,m_3}(x,y)) + R_{m_2,m_3}(D_{m_2,m_3}(x,y))$$

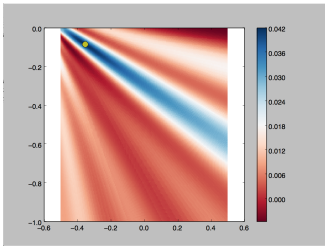
,where $R_{m_1,m_2}(\tau)$,$R_{m_1,m_3}(\tau)$, and $R_{m_2,m_3}(\tau)$ denote GCC output from microphone pairs m_1m_2, m_1m_3, and m_2m_3.

Liklihood maps from two arrays can be combined into final liklihood map:

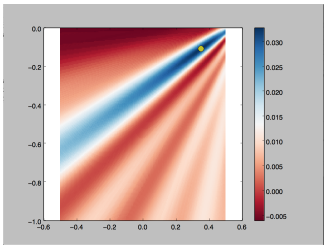
$$L(x, y) = L_1(x, y)L_2(x, y) \tag{6}$$

, where $L_1(x, y)$ and $L_2(x, y)$ represents the liklihood map from array 1 and array 2.

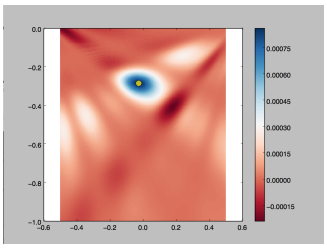
To see the effect of accuracy improvement using multiple arrays, fig 6 shows a real life localization where the source is placed at $(0 \text{ cm}, -30 \text{ cm})$. It shows the individual likli-hood map from each array and also the combined liklihood map according to equation 6. Individual array gives accurate angle estimate, but has high uncertainty in source distance. It demonstrated that by combining estimates from two arrays the system is able to perform more accurate localization.



(a) localization with only array 1



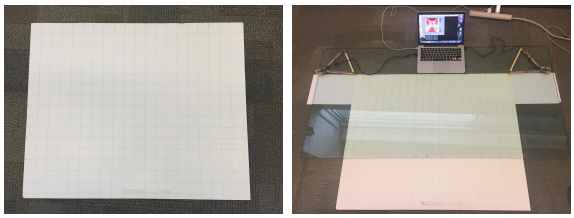
(b) localization with only array 2



(c) localization with both arrays

Fig. 6: Liklihood maps for localizating point at $(0.0, -0.3)$ m

From a timing point of view, the micro-controller spends 12 millisecond on sampling microphone data before sending the data to the computer for processing. Sending data through USB port takes another 18 millisecond, and processing on computer takes around 50 millisecond. Therefore, the totoal time lag between sound source and localization is around 80



(a) 1 meter by 1 meter grid (b) array placement

Fig. 7: Setup for localization accuracy testing

millisecond.

V. Experiment

A. Setup

1) Point localization: To test localization accuracy, an one meter by one meter grid was set up and the arrays are placed at the top left and top right corners of the grid. Fig 7 shows a picture of the setup. A total of 32 positions are chosen uniformly in this region where microphone data is recorded.

2) Movement tracking: To test how well the arrays track movement, we mounted a rotating disk 40 centimeter in diameter onto the grid at $(x = 0 \text{ cm}, y = -30 \text{ cm})$. Fig 11 shows a picture of the sestup. A sound source is placed on the edge of rotating disk and the arrays localizes the sound source as it rotates in a circle. In this experiment, we tested how accuracy changes with:

- different window sizes
- different audio sources
- different movement tracking filters
- different movement speeds

To test how different sound sources impact localization quality, we conducted three experiments on the same movement track with three different sound sources:

White Noise	A recording of white noise.
Music A	A music that has normal audio amplitude throughout experiment period was chosen. "Honest Eyes" by Black Tide was the music used.
Music B	A music with intermittent low amplitude sections was chosen. "Canon" was the music used.

To test how sound source movement speed affects localization quality, each of three experiments were conducted at two different speeds:

Normal	An angular speed of 0.5 rad/s was maintained, which translates to linear speed of 10 cm/s.
Fast	An angular speed of 1.0 rad/s was maintained, which translates to linear speed of 20 cm/s.

For each experiment conducted, two different movement filters are tests:

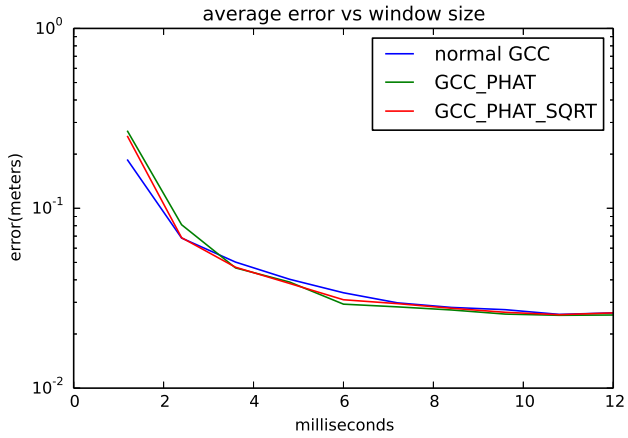


Fig. 8: accuracy versus window size

Averaging filter	localization for past 0.5 seconds are averaged and outputed as current estimate.
Kalman filter	A 2nd order Kalman filtering is used.

B. Results

1) Point localization: To test how accuracy varies with window size, the algorithm is fed with recorded microphone data of different window size. Fig 8 shows how accuracy changes with window length for three GCC algorithms. The error decreases as window size increases and plateaus after window size exceeds around 10 millisecond. The lowest error achieved is 2.53 centimeters. It is achieved when window size is set to 12 millisecond and GCC_PHAT is used for TDOA estimation.

Although accuracy improves with window length, the calculation time also increases with window length. The part of calculation that depends on window length is using cross correlation to estimate TDOA. cross correlation can be calculated with FFT and the runtime is of order $O(N \log N)$. We measured how the computation time varies with window length and Figure 9 shows the result. The runtime increases approximately linearly in the window size region of interest.

We also calculated the localization error for each tested point in the region. Figure 10 shows a heatmap of the error distribution inside the grid. The error is below 3 cm for most areas inside the region. There is one error spike in the mid-left region and we contribute this to audio source misplacecment becuase the error is fairly low and consistent around that spike region.

2) Movement tracking: Fig **??** gives an intuitive representation of how accuracy changes with window size. When window size is small(1.02 millisecond), the audio does not contain enough information to reliably estimate TDOA. The localization is noisy. As window size increases, the localization converges to the shape of ground truth circle. Fig 13 shows how the error changes with window size. The general trend is similar to that in point localization case. The error decreases as window size increases and plateaus after window size exceeds around 10 milliseconds.

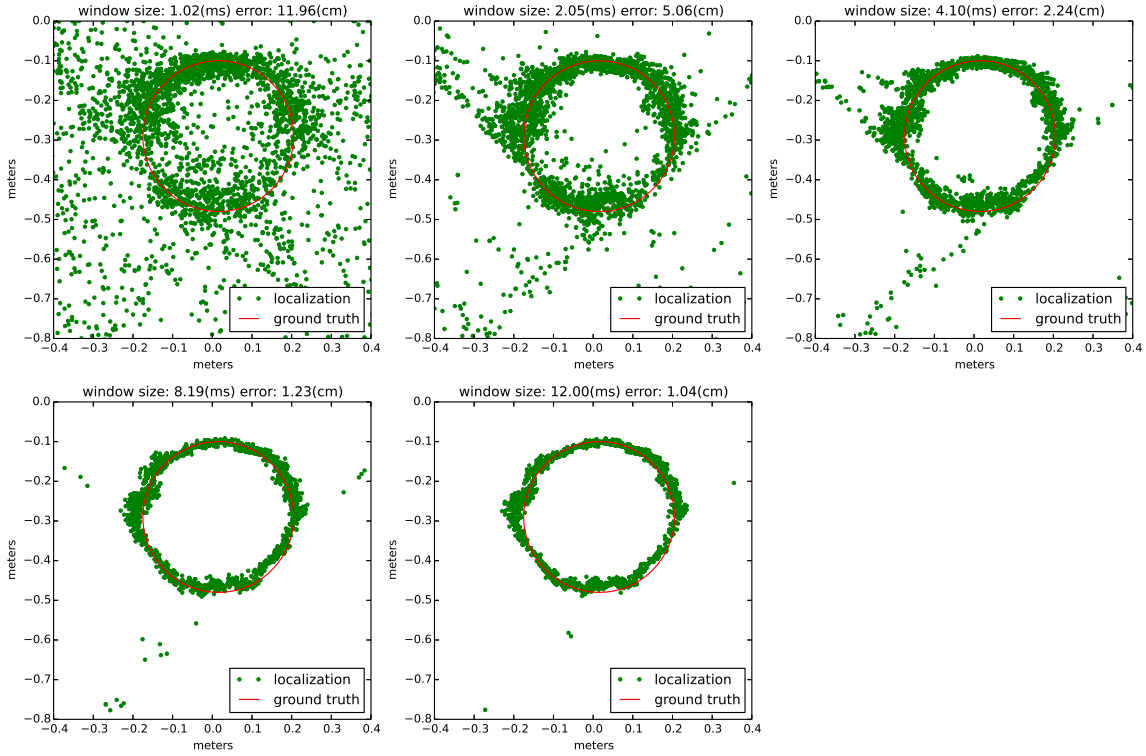


Fig. 12: Localization quality versus window size

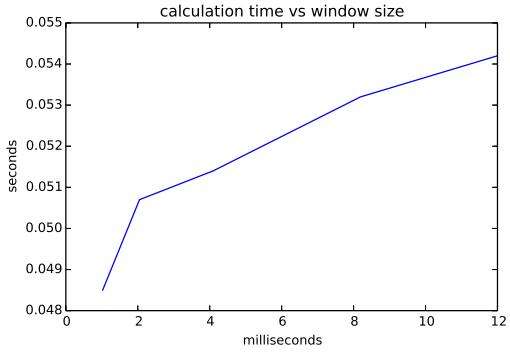


Fig. 9: speed versus window size

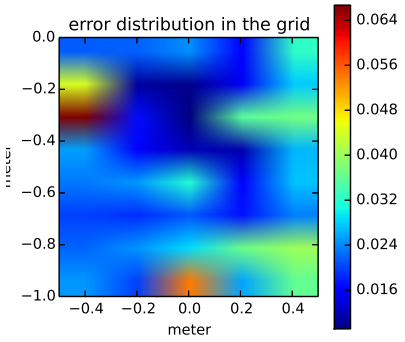


Fig. 10: error distribution in the grid

Fig 14 to 16 shows results for experiments at normal speed, and Fig 17 to 19 shows results at fast speed. By comparing localization error for each audio source between normal movement speed and fast movement speed, we find the localization error does not depend on how fast the sound sorce is moving. For example fig 15 and fig 18 shows that localization error is 1.289 cm at normal movement speed and 1.291 cm at fast movement speed.

For normal speed movement tracking, localization error is

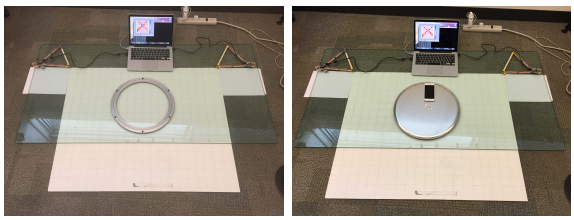


Fig. 11: Setup for circle movement localization

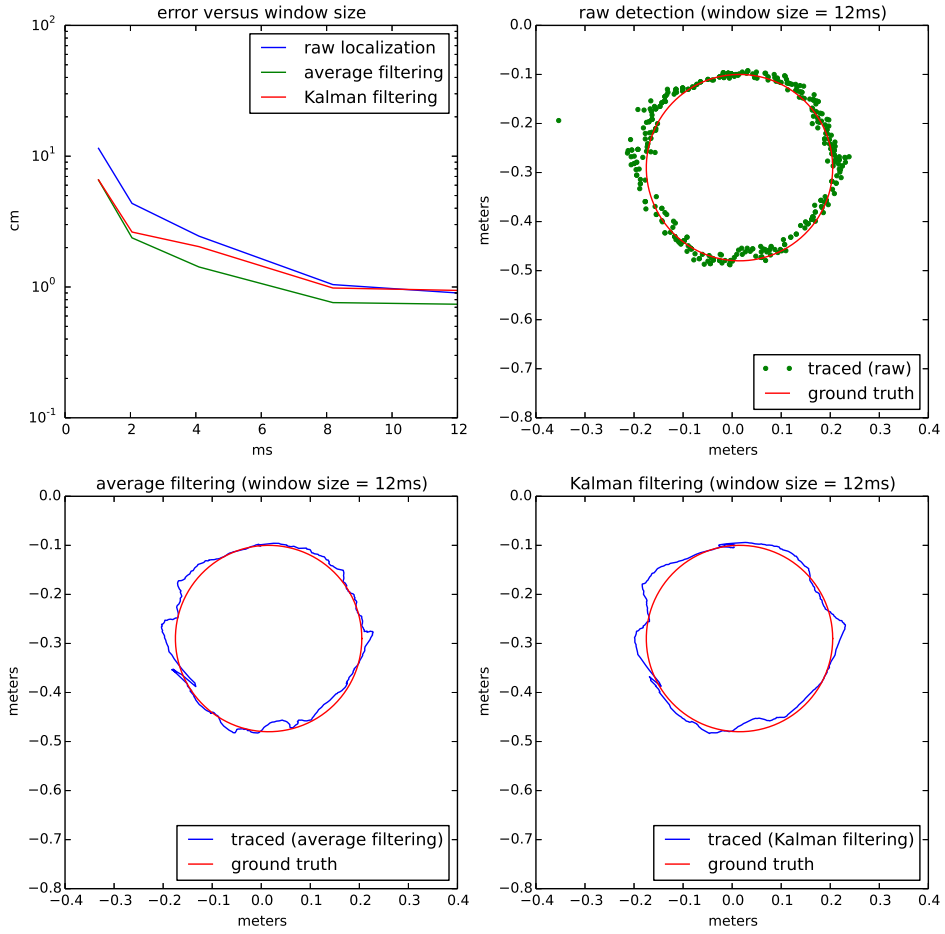


Fig. 14: white noise (10 cm per second)

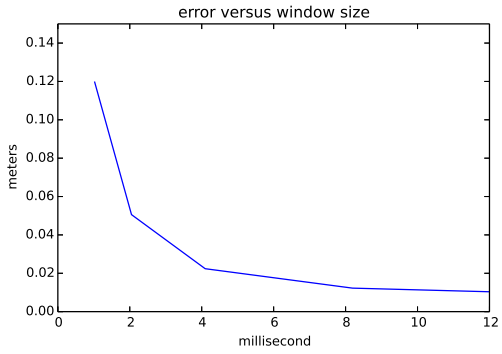


Fig. 13: Localization error versus window size

0.9 cm for white noise, 1.29 cm for Music A, and 2.9 cm for Music B. Localization accuracy is the best for white noise source, and worst for Music B. This is consistent with our expectation because low amplitude regions in Music B would cause arrays to lose track of where the source is. This can also be seen from the "blank" regions in fig 16.

It also shows that raw detection has the most amount of jiggling. Kalman filter reduces the amount of jiggling from raw detection. Averaging filter has the least amount of jiggling. However, averaging filter averages detection outputs from past 0.5 seconds, which makes the filtered output lag the real movement.

VI. Conclusion

In this paper, we proposed and built an inexpensive, portable yet reasonably accurate sound localization system with two microphone arrays. We analyzed different array achitectures and demonstrated that a two array achitecture achieves better accuracy compared to single array system of similar size. Using cross-correlation output as liklihood for

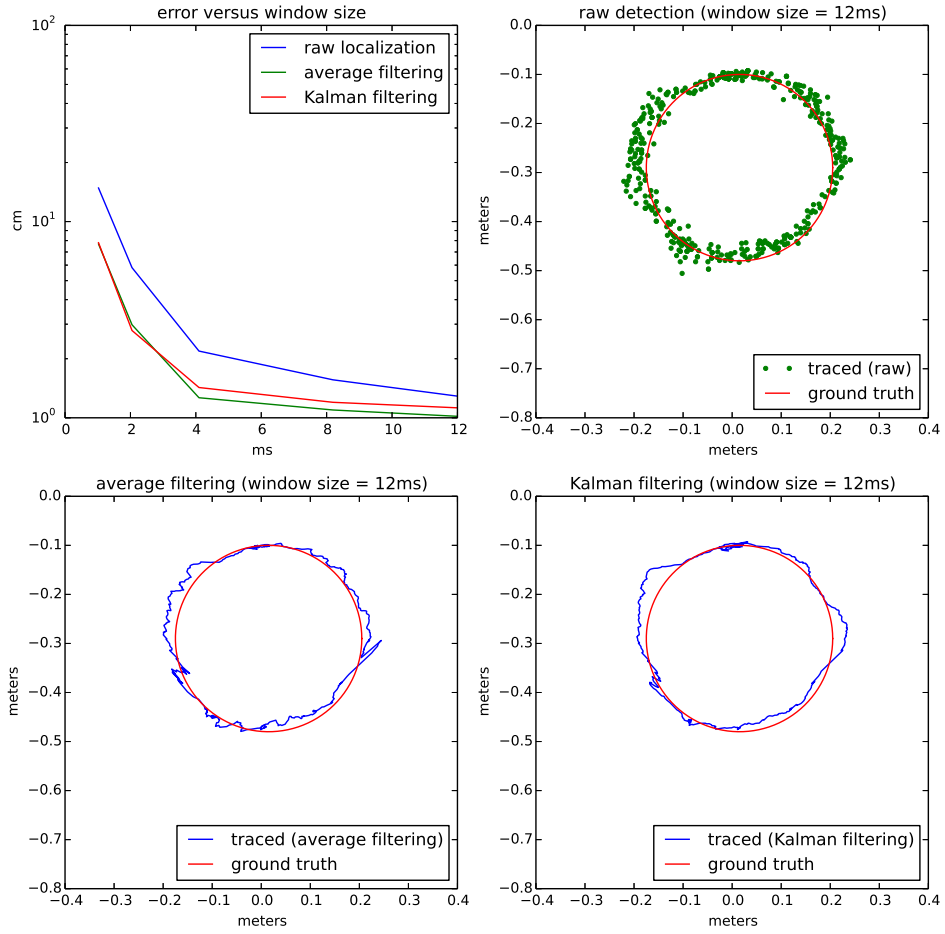


Fig. 15: music A (10 cm per second)

each input delay, each array outputs a heatmap of liklihood for each possible location in the area. We demonstrated that by merging two heatmaps from two arrays, we were able to achieve good localization accuracy (less than 3 cm) in a local one meter by one meter region.

This system can be used in HCI applications that uses sound position and movement as input, such as drawing with music. Another application is to build AI games with physical pieces where the computer controls some of the pieces. This system can also be used to build Augmented Reality (AR) applications that allow user computer interaction with music tags.

For future directions, this system can be extended with another microphone on each array to perform 3D localization. Another possible direction is to investigate multisource localization.

REFERENCES

[1] Wing, Michael G. and Eklund, Aaron and Kellogg, Loren D. "Consumer-Grade Global Positioning System (GPS) Accuracy and Reliability," *Journal of Forestry*, 2005.

[2] Bekkelien, Anja, Michel Deriaz, and Stphane Marchand-Maillet. "Bluetooth indoor positioning," *Master's thesis, University of Geneva*, 2012.

[3] Hui Liu; Darabi, H.; Banerjee, P.; Jing Liu, "Survey of Wireless Indoor Positioning Techniques and Systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 2007

[4] Yazici, A.; Yayan, U.; Yucel, H., "An ultrasonic based indoor positioning system," *2011 International Symposium on Innovations in Intelligent Systems and Applications (INISTA)*, 2011

[5] Chowdhury, T., Aarabi, P., Weijian Zhou, Yuan Zhonglin, and Kai Zou, "Extended touch user interfaces," *IEEE International Conference on Multimedia and Expo (ICME)*, 2013.

[6] Pham, D. T., et al. "Tangible acoustic interface approaches," *Proceedings of IPROMS 2005 Virtual Conference*, 2005.

[7] XueXin Yap, Khong, A.W.H., and Woon-Seng Gan, "Localization of acoustic source on solids: A linear predictive coding based algorithm for location template matching," *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, 2010

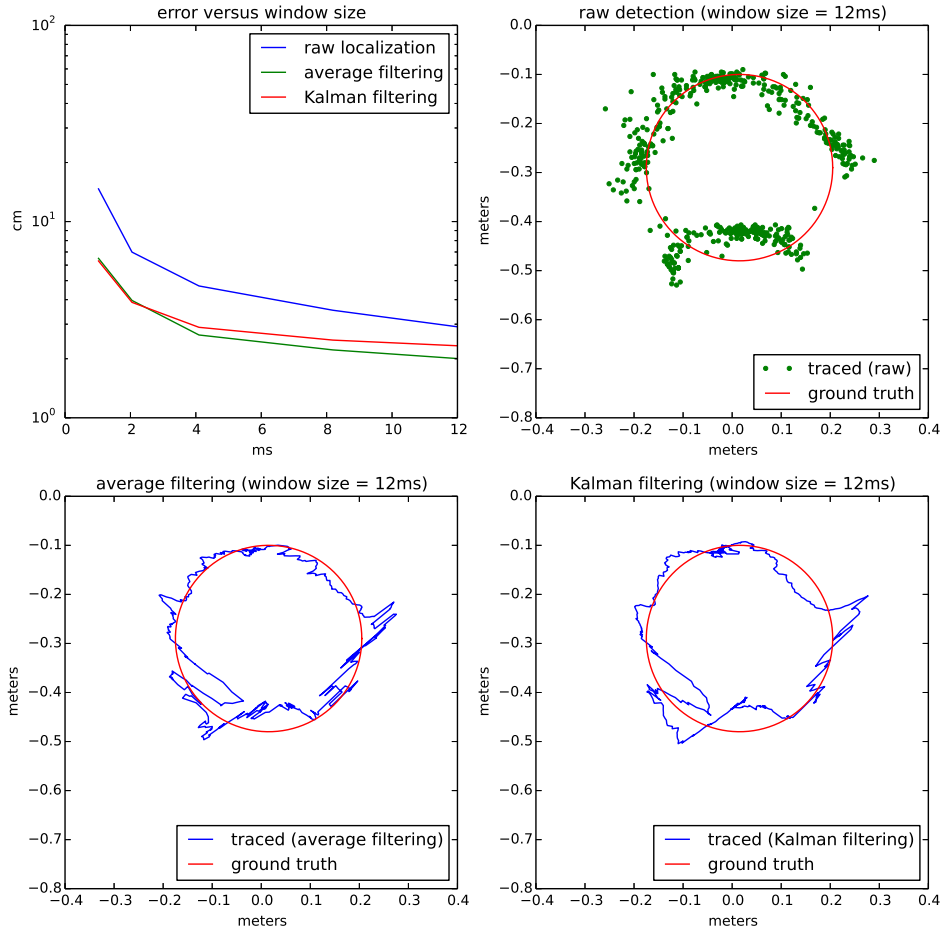


Fig. 16: music B (10 cm per second)

[8] Chowdhury, Tusi. "Single Microphone Tap Localization," *University of Toronto*, 2013.

[9] Ishii, Hiroshi, et al. "PingPongPlus: design of an athletic-tangible interface for computer-supported cooperative play," *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*. ACM, 1999.

[10] Paradiso, Joseph A., et al. "Passive acoustic sensing for tracking knocks atop large interactive displays," *Proceedings of IEEE. Vol. 1. IEEE*, 2002.

[11] MicLoc, URL: http://ruralhacker.blogspot.ca/p/micloc.html

[12] Polotti, Pietro, et al. "Acoustic Localization of Tactile Interactions for the Development of Novel Tangible Interfaces," *Proc. of the 8th Int. Conference on Digital Audio Effects (DAFX-05)*, 2005.

[13] Paradiso, Joseph A., et al. "Sensor systems for interactive surfaces," *IBM Systems Journal*, 2000.

[14] Checka, Nisha "A system for tracking and characterizing acoustic impacts on large interactive surfaces," *Diss. Massachusetts Institute of Technology*, 2001.

[15] Brandstein, M.S.; Silverman, H.F., "A robust method for speech signal time-delay estimation in reverberant rooms," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1997.

[16] Knapp, C. and Carter, G.Clifford, "The generalized correlation method for estimation of time delay," *IEEE Transactions on Acoustics, Speech and Signal Processing*, 1976.

[17] Aarabi, P. and Guangji Shi, "Phase-based dual-microphone robust speech enhancement," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 2004.

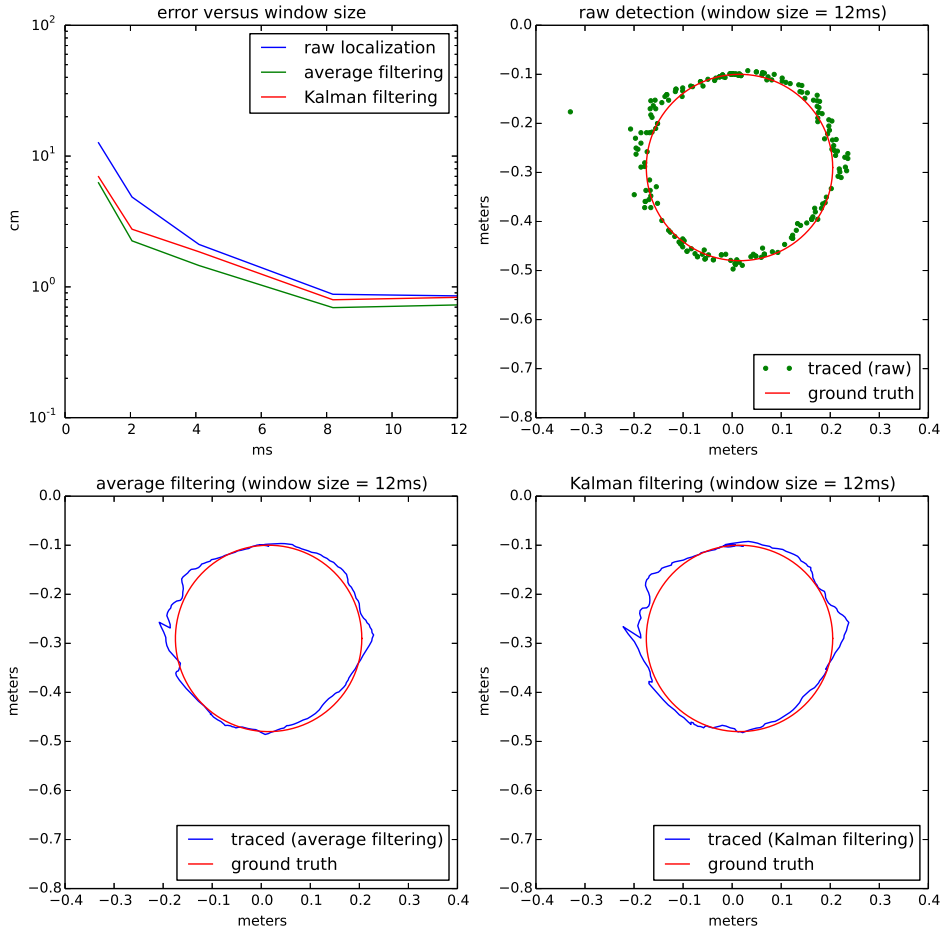


Fig. 17: white noise (20 cm per second)

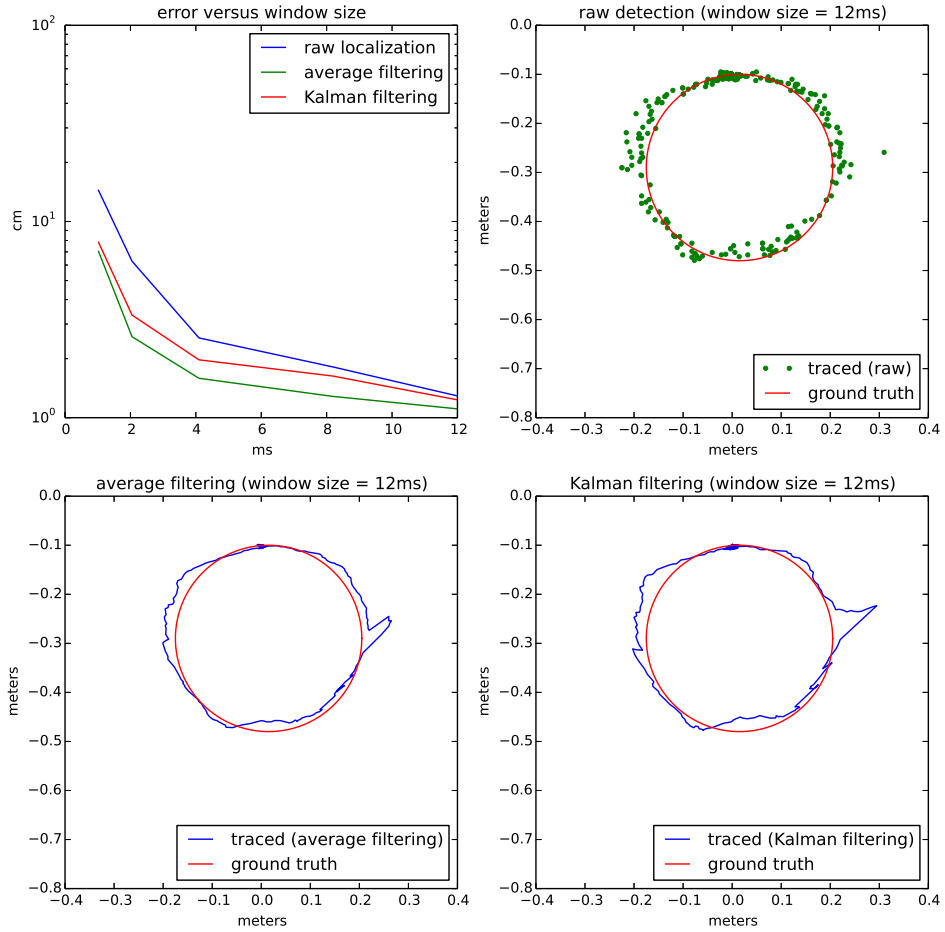


Fig. 18: music A (20 cm per second)

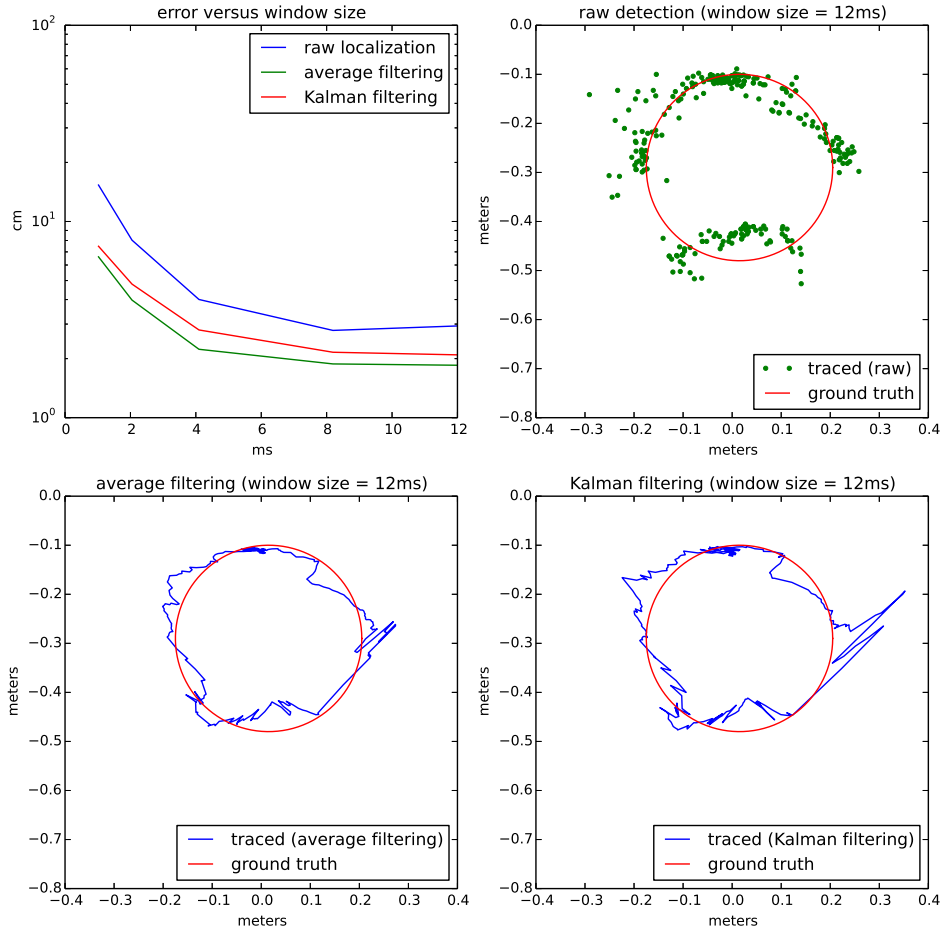


Fig. 19: music B (20 cm per second)