

# Paint with Music

XXX  
XXX

XXX  
XXX

**Abstract**—The abstract goes here.

## I. INTRODUCTION

## II. PRIOR WORK

## III. SOUND LOCALIZATION

Points with the same time difference of arrival(TDOA) to two fixed points on a plane form a hyperbola. Since TDOA from each pair of microphones gives a hyperbola in the plane, localization becomes finding intersection of hyperbolas when more than two microphones are used. Localization relies on accurate estimate of delay differences between microphones.

Generalized Cross Correlation(GCC) provides a framework to estimate delay differences  $t_0$  between two signals  $x_1(t)$  and  $x_2(t)$ :

$$t_0 = \arg \max_{\tau} R_{x_1 x_2}(\tau) \quad (1)$$

$$R_{x_1 x_2}(\tau) = \int_{-\infty}^{\infty} W(\omega) X_1(\omega) X_2^*(\omega) e^{j\omega\tau} d\omega \quad (2)$$

, 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 cross correlation estimator. We experimented with three ways of prefiltering the signal:

GCC

$W(\omega) = 1$ . No prefiltering is done. This is normal cross correlation.

GCC\_PHAT

$W(\omega) = \frac{1}{|X_1(\omega)X_2^*(\omega)|}$ . Each frequency is divided by its magnitude. Only phase information contributes to delay estimation.

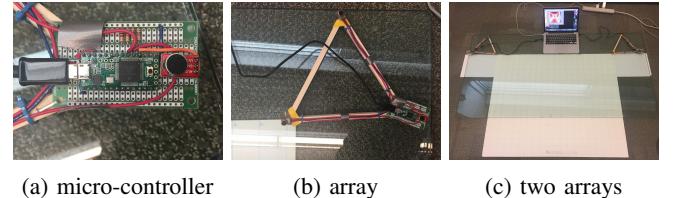
GCC\_PHAT\_SQRT

$W(\omega) = \frac{1}{|X_1(\omega)X_2^*(\omega)|^{0.5}}$ . This is somewhere between GCC and GCC\_PHAT. part of magnitude information is included in delay estimation.

## IV. SYSTEM

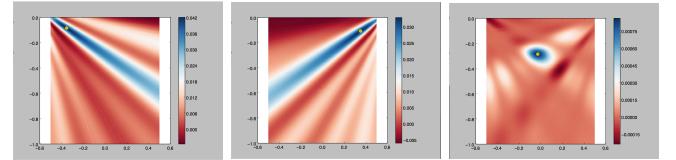
The system uses two arrays, each with three microphones mounted on vertices of a 20cm equilateral triangle. A micro-controller is also attached to one of the vertices. Fig 1 shows a picture of the array setup. Micro-controller used in this project is *teensy 3.1*. It has 64k RAM memory and the ADC is 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 for localization.

To speed up computation for real time localization, instead of searching for  $t_0$  that maximizes equation 1, a grid search



(a) micro-controller      (b) array      (c) two arrays

Fig. 1: Localization arrays



(a) localization with only array 1      (b) localization with only array 2      (c) localization with both arrays

Fig. 2: Likelihood maps for localizing point at  $(0.0, -0.3)$  m

in 2D grid is performed. For each point in the grid, the TDOA to each microphone pair can be precomputed. Then localization resolves to calculating GCC for each microphone pair and performing a lookup for each point in the grid. To further improve localization accuracy, instead of using point estimate for TDOA, a likelihood map is built. Each entry in GCC output is used as the likelihood for that delay. With three microphones  $m_1, m_2, m_3$ , there are three microphone pairs:  $m_1m_2, m_1m_3, m_2m_3$ . 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 likelihood map can be built as:

$$L(x, y) = R_{m_1, m_2}(D_{m_1, m_2}(x, y)) + R_{m_1, m_3}(D_{m_1, m_3}(x, y)) + R_{m_2, m_3}(D_{m_2, m_3}(x, y)) \quad (3)$$

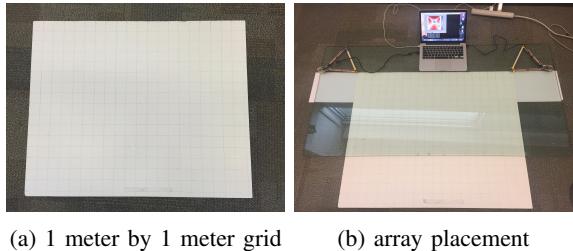
, where  $R_{m_1, m_2}(\tau), R_{m_1, m_3}(\tau)$ , and  $R_{m_2, m_3}(\tau)$  denote GCC output from microphone pairs  $m_1m_2, m_1m_3$ , and  $m_2m_3$ .

Likelihood maps from two arrays can be combined into final likelihood map:

$$L(x, y) = L_1(x, y)L_2(x, y) \quad (3)$$

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

To see the effect of using multiple arrays, fig 2 shows the individual likelihood map from each array and also the combined likelihood map according to equation 3. Individual array gives accurate angle estimate, but has high uncertainty



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

Fig. 3: Setup for localization accuracy testing

in distance estimate. By combining estimates from two arrays, the angle estimate can be effectively combined to estimate distance.

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 total time lag between sound source and localization is around 80 millisecond.

## V. EXPERIMENT RESULTS

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 3 shows a picture of the setup.

A total of 32 positions are chosen uniformly in this region where microphone data is recorded. To test how accuracy varies with window size, the algorithm is fed with recorded microphone data with different segment length. Fig 4 shows how accuracy changes with window length for three GCC algorithms. The error lowers 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 5 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 6 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 placement error because the error is fairly low and consistent around that spike region.

To test how well the arrays track movement, we mounted a rotating disk 40 centimeter in diameter onto the grid at  $(x = 0, y = -0.3)$ . Fig 7 shows a picture of the setup. A sound source is placed on the edge of rotating disk and the

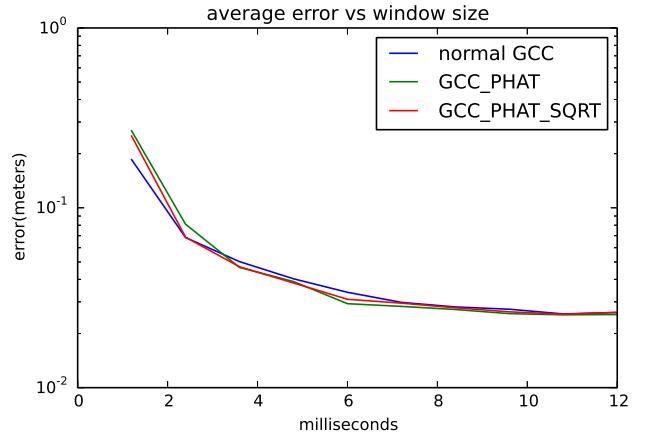


Fig. 4: accuracy versus window size

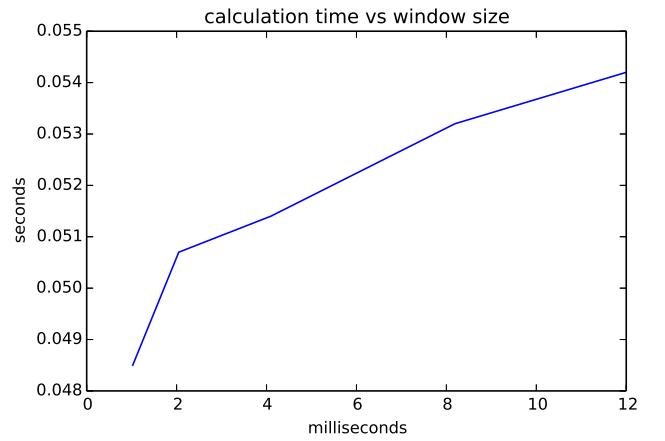


Fig. 5: speed versus window size

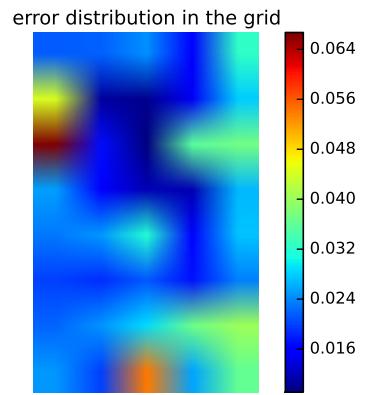


Fig. 6: error distribution in the grid

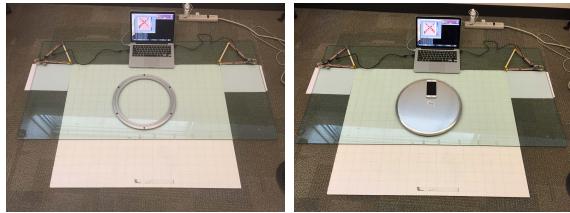


Fig. 7: Setup for circle movement localization

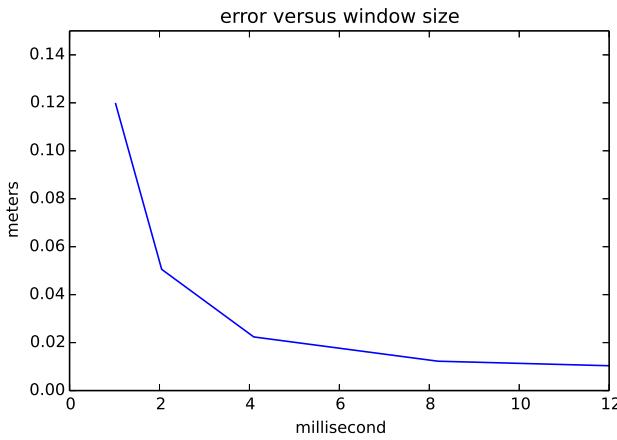


Fig. 9: Localization error versus window size

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

Fig 9 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 9 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.

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:

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

Fig 10 shows results for experiments with at normal speed, and Fig 11 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 source is moving. For example fig 10b and fig 11b shows that localization error is 1.289 cm at normal movement speed and 1.291 cm at fast movement speed.

From fig 10, localization error is 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 make arrays lose track of where the source is. This can also be seen from the "blank" regions in fig 10c.

Fig 10 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

The conclusion goes here.

## ACKNOWLEDGMENT

The authors would like to thank...

## REFERENCES

- [1] H. Kopka and P. W. Daly, *A Guide to L<sup>A</sup>T<sub>E</sub>X*, 3rd ed. Harlow, England: Addison-Wesley, 1999.

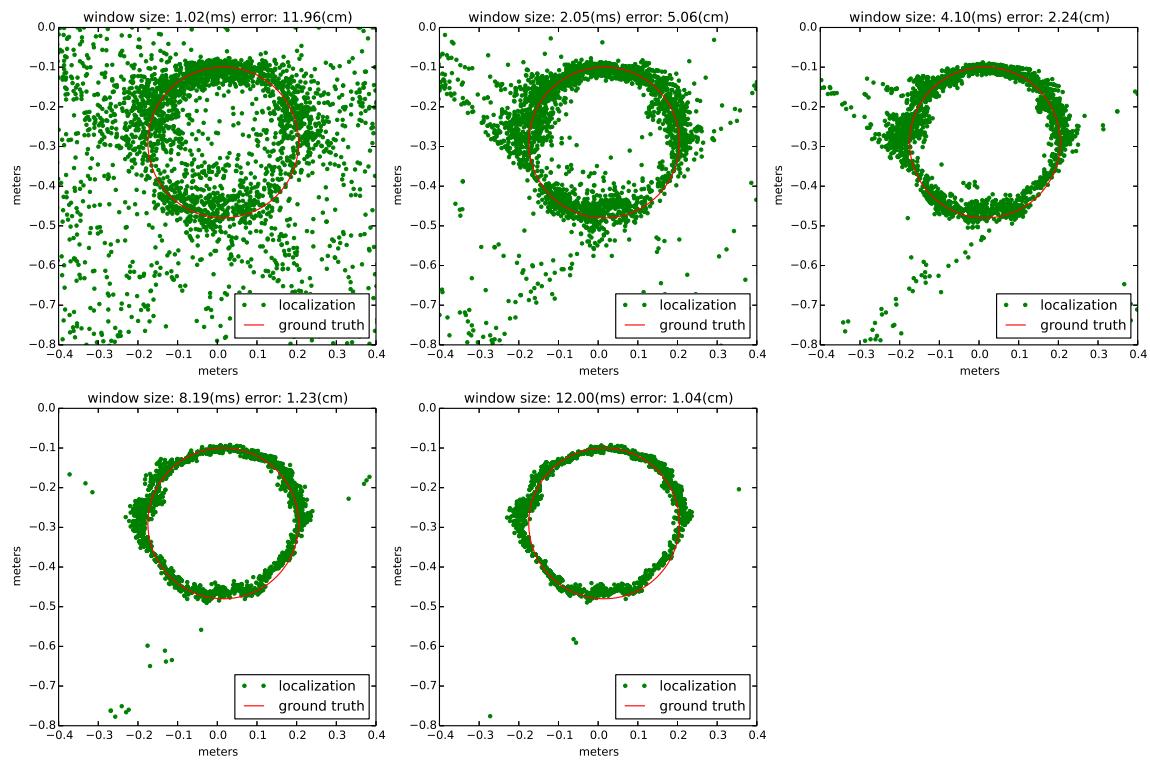
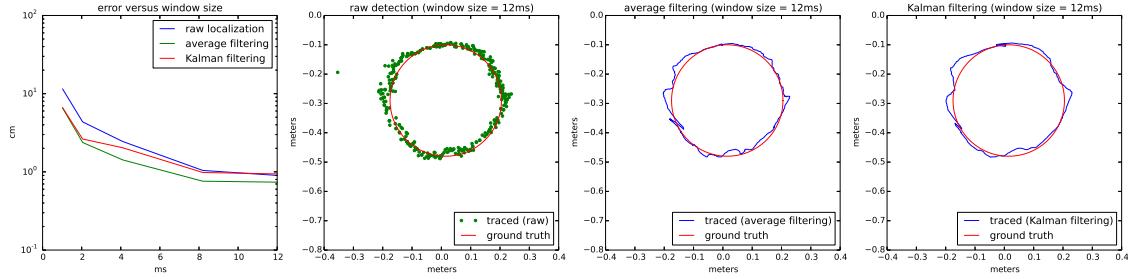
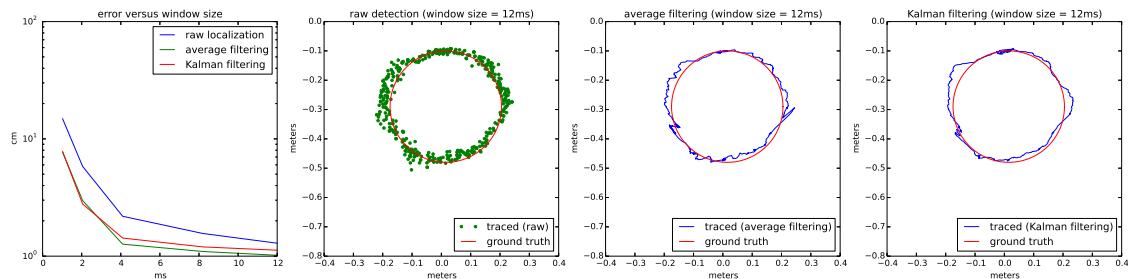


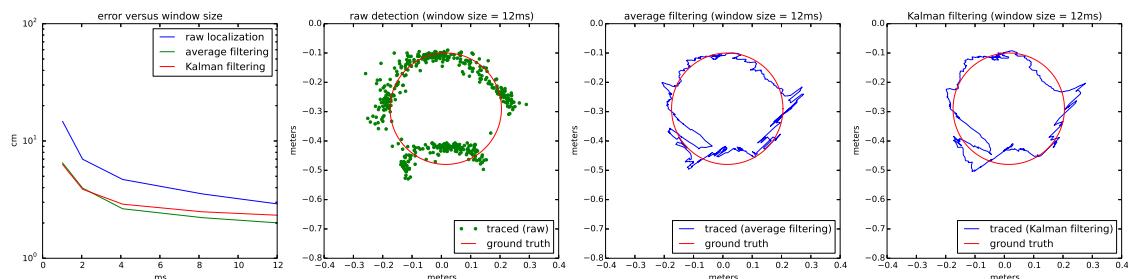
Fig. 8: Localization quality versus window size



(a) white noise

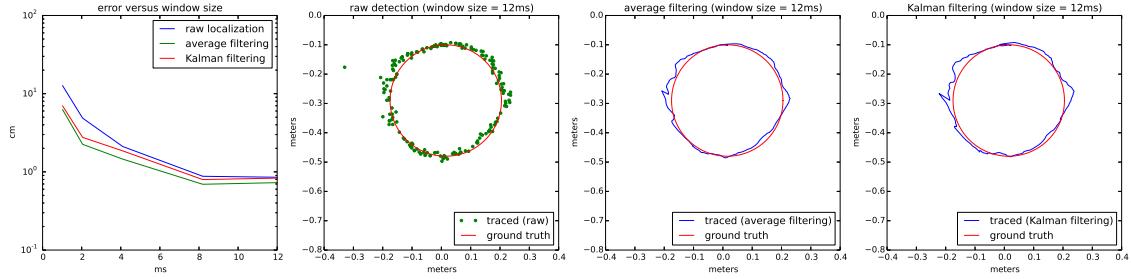


(b) music A

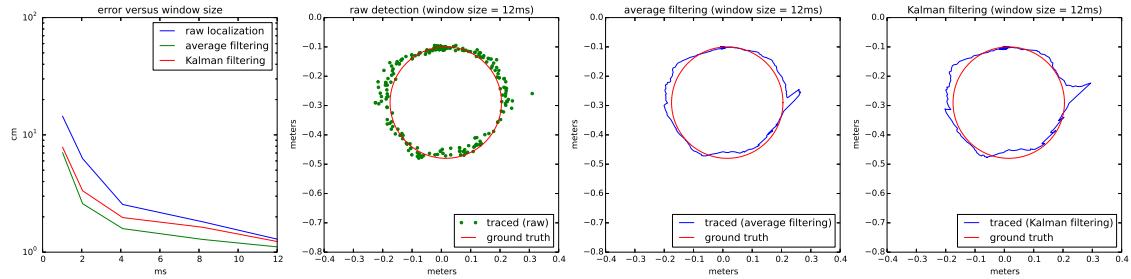


(c) music B

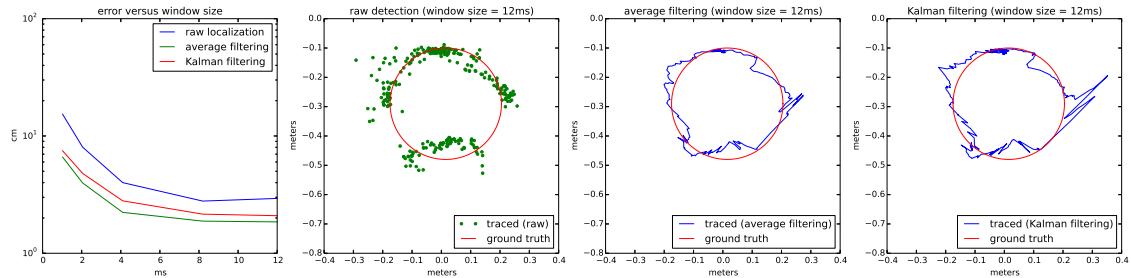
Fig. 10: Localization of circle movement with different sound sources. Sound source is moving at 10 cm per second



(a) white noise



(b) music A



(c) music B

Fig. 11: Localization of circle movement with different sound sources. Sound source is moving at 20 cm per second