Location and Tracking Based on Acoustic Signal

王天麒 11711613

叶冠辉 11712421

成为 11712319

ORIGINAL OBJECTIVE

Most device-free gesture tracking uses visual techniques, however acoustic signal is a low-energy technique for tracking and location but not used widely. Further more, acoustic speaker and microphone are low-cost and widely used devices. There are some solutions of device-free gesture tracking, such as Google's Soli system [1] which uses customized hardware to generate 60GHz wireless signal to track small movement. Rajalakshmi Nandakumar's FingerIO [2] achieve this by transforming the device into an active sonar system that transmits inaudible sound signals and tracks the echoes of the finger at its microphones. Because audio signal propagates slowly, it is possible to achieve high accuracy [2] Many smart phone supports two speakers for stereo playback, without any extra device it has potential to do tracking and location. CHEN H, LI F and WANG Y raise EchoTrack [3] system to implement device-free tracking without external hardware.

Referring to others studies, we have learned some advantages and disadvantages of device-free tracking based on sound. With these feature, sound base tracking has wide application scenario, such as extend IO method, VR/AR applications, phone-to-phone location [4] etc. In this semester, We are objecting to implement a device-free hand tracking and location with Commercial-Off-The-Shelf mobile device(smart phone) and develop a simple distance and location measurement application on it. By implementing this technology, we hope to provide guidance for our future work.

WORK DESCRIPTION

Our work is based on EchoTrack [3], a device-free hand tracking solution, which can be enable on dual channel smart phone. It is capable of continuous localization resolution within mean 3cm of 76% and 2 cm of 48% [3].

At preliminary stage, we use Matlab to analysis the feature of recording sound and complete configuration of measurement arguments. The future plan is to turn the Analysis step to low-level code in smart phone to implement measure distance and tracking. At the end of this semester we will make a android application prototype.

There are several stages in this work:

Implement hand track with EchoTrack method, steps show in Fig.1

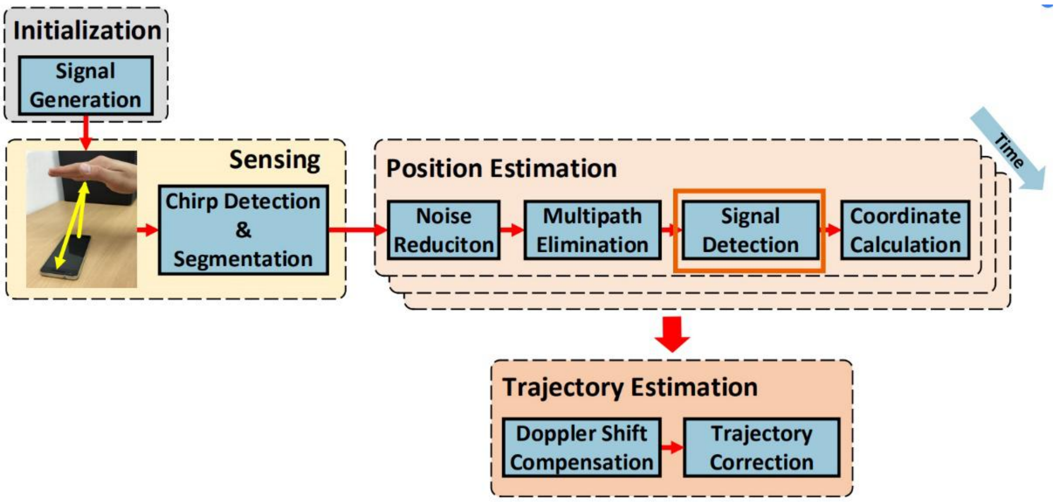


Fig.1 steps of EchoTrack method [3]

After we can get stable and correct measurement, integrate it in an android application.

CURRENT PROGRESS

In detail, our work has finished following stages:

Initialization stage:

Choose a smart phone which support dual channel sound(our experiment using HTC M9ew) and test the which channel the two speakers are corresponded to left/right channel.

Then according to EchoTrack method, we generate two detecting chirp, left chirp is short signal with increased frequency and right chirp is decreased frequency. Because of the device limitation [3] chirp frequency range is [23kHz, 16kHz]. Our smart phone's microphone has 48kHz sampling rate, so we choose 48kHz as chirp generate sampling rate.

Generate two chirp signals:

$$chirp_{left} = \cos(\omega_t t), \omega_t = \frac{f_t - f_s}{t_0} \cdot t + f_s$$

$$chirp_{right} = \cos(\omega'_t t), \omega'_t = \frac{f_s - f_t}{t_0} + f_t$$

$$f_s = 23k, f_t = 16k$$

Right chirp has a latency before left chirp emit.

Detail information in EchoTrack [3] page 2 to 3.

Our original two channel sound as shown in Fig.2

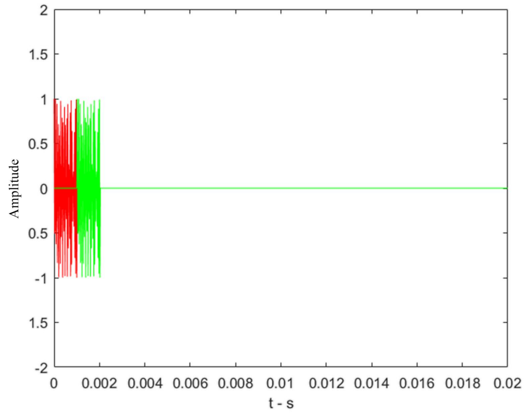


Fig.2: dual channel chirp, left chirp: red, right chirp: green

Each chirp sustained 1ms and period is 20ms. In experiment, the sound is played repetitively.

Sensing stage:

Set smart phone's microphone record continually while the speakers send ready-make chirp sound. Then devide recording sound into several fragments including original chirps and its echo.

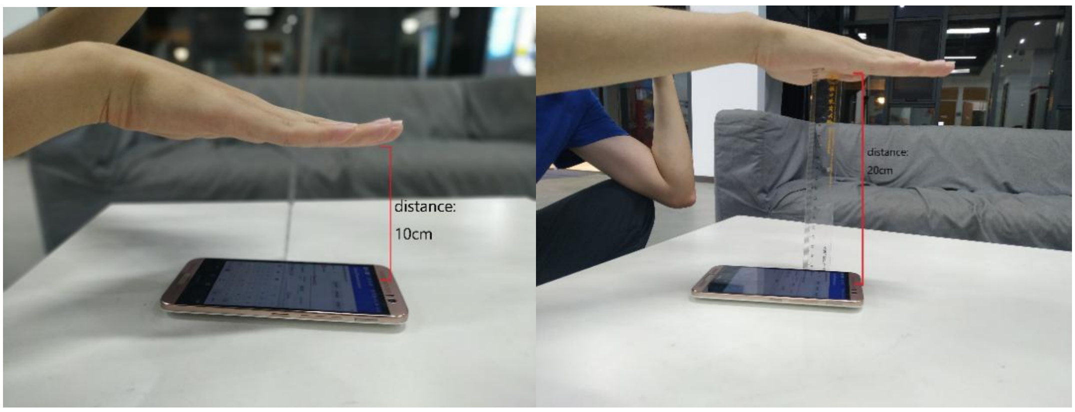


Fig.3

We chose a relatively quiet and empty environment to measured two sets of data, one set at a distance of 10 centimeters and the other set at a distance of 20 centimeters.

Position Estimation Stage

Before experiment, we make a simulation environment to analysis signal behavior.

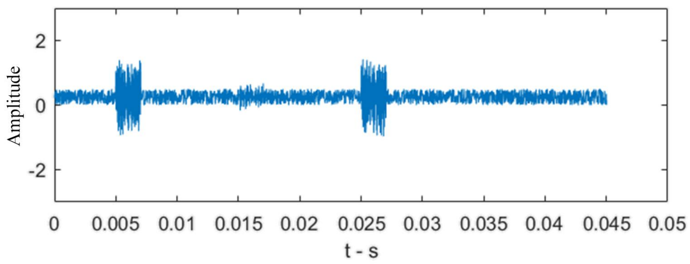


Fig.4

Noise reduction:

Firstly, chirp's echos are hiding in environment noise, so the sound should process though band-pass filter which reserve 16kHz to 23kHz. After that the most of noise has been eliminate. Fig.5 (a)(b) shows the original sound and sound though band-pass filter.

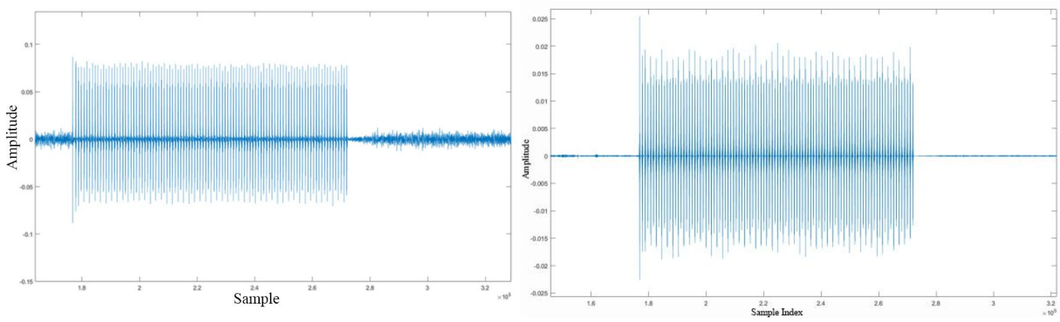


Fig5.(a)

Fig5.(b)

Envelope detection:

To detect chirp and its echos, the signal need to pass match filter and get its envelope.

In this step, we have two ways to find the places which chirps and its echos occur.

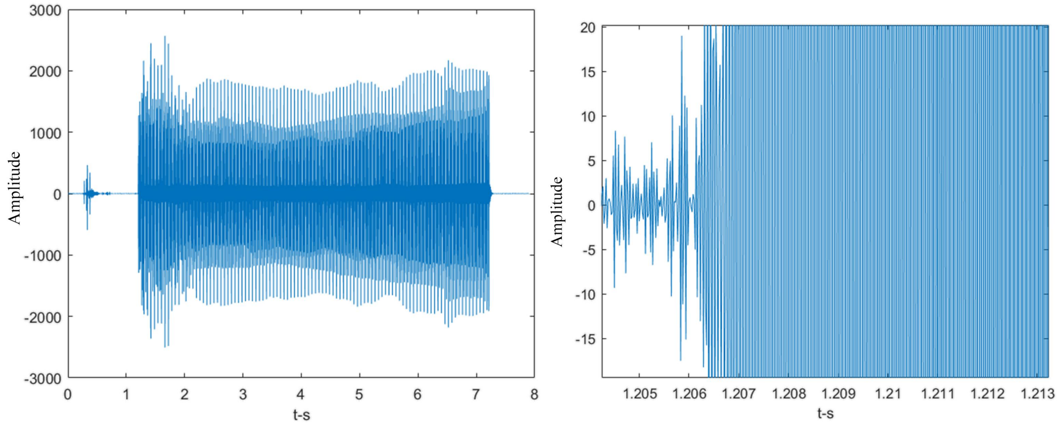
The first is using match filter as just mentioned.

$$\mathcal{E}(p, s) = \mathcal{F}^{-1}\left\{\mathcal{Z}\left\{\overline{\mathcal{F}\{p\}} \cdot \mathcal{F}\left\{s\right\}\right\}\right\}$$

p and s represent chirp and recorded signal [3]

The second method is dechirp.

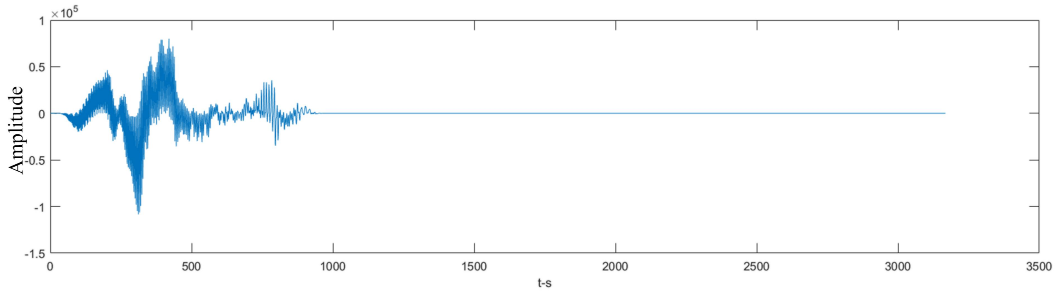
We extract a sliding window length 20ms and step length 2ms. In each window we apply dechirp of windowed signal with original chirp of length 20ms to check if we get the first highest peak.

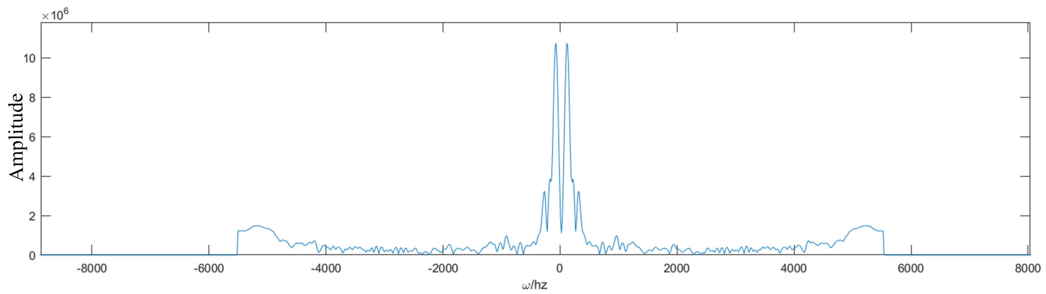


After we finish Line up, we try to calculate window by window to dechirp the original signal finding the echo distant.

In this approach, we basically aware that our original emitted signal is composite of 20ms up chirp, 20ms down chirp and 20ms blank, so according to the paper we set the window size on 66ms with 10% overlap between windows.

Finally, we detect the highest 2 peaks on frequency domain and calculate their frequency difference.





In this part, we can clearly observed that there are some echo peaks beside the main peak. According to our detecting program we can give out the frequency difference between those 2 peaks is about 100hz.

Trajectory Estimation Stage:

Using compensation method to offsetting the error due to doppler shift and perform a trajectory correction method for improving trajectory accuracy [3].

Application Stage:

Implement the measurement calculation in android application.

These two stages we haven't start yet.

CURRENT DIFFICULTIES

1. Signal processing during long time. Because we aim to implement a real-time distance measurement application, too long a delay is not allowed.

2. Measurement accuracy cannot ensure, EchoTrack only assure 2 to 3cm pecision. We are consider using OFDM [2] to improve Measurement accuracy.

3. For different smart phone the distance between speakers and microphone is also different. So software portability also is a big question here.

4. The measurement processing part may be a difficult task to move into a smart phone.

WORK DIVISION

王天麒: Report writing, simulation, process received signal

成为: Modify report, generate chirp signal, process received signal

叶冠辉: Make presentation PPT, Modify report, process received signal

REFERENCES

[1] Google project soli. https://www.google.com/atap/project-soli/.

[2] Rajalakshmi Nandakumar, Vikram Iyer, Desney Tan, and Shyamnath Gollakota. FingerIO: Using active sonar forfine-grained finger tracking. In Proc. ACM CHI, 2016.

[3] CHEN H, LI F, WANG Y. Echotrack: Acoustic device-free hand tracking on smart phones[C]//IEEE INFOCOM 2017-IEEE Conference on Computer Communications. [S.l.]: IEEE, 2017: 1-9.

[4] Chunyi Peng, Guobin Shen, Yongguang Zhang, Yanlin Li,and Kun Tan. Beepbeep: a high accuracy acoustic ranging system using COTS mobile devices. In Proc. ACM SenSys, 2007.