EchoSight-Pro

User Guidance

For version 0.1.0

Author: Zhiqiang Jiang(Z.Jiang)

User Guidance Index

0. Preface

0.1 What is EchoSight-Pro

EchoSight-Pro is a medical ultrasound signal chain core framework, which is designed and developed by myself alone.
The design of EchoSight-Pro is quite engineering oriented and efficiency targeted.

0.2 Who can benefit from EchoSight-Pro

Theoretical Research People
For the Master and PhD students, as well as professors at universities or research institutions, you can use EchoSight-Pro to check and evaluate your research topics.

Ultrasound Start-up Corporation People
Creating an ultrasound machine from scratch is always of tons of uncertainties and big risks. You can use EchoSight-Pro to evaluate your hardware prototype by dumping your raw data before machine is built up. And it can give you confirmed answer to whether your industry computer is able to support software beam forming or not.
Also, EchoSight-Pro can be used to check if the channel data from your hardware board/FPGA is correct or not.

Traditional Ultrasound Corporation People
Two approaches of ultrasound commercial framework design exist at least. One is FPGA / hardware chip based; another is software based.
EchoSight-Pro can be used to testify the software RTB synthetic beam forming possibility, before your team starting to explore the new technique road path, and check your front end before you preparing project resource and recruit the team.

0.3 Copyrights and charges

EchoSight-Pro is free of charge to use, as long as you do **NOT** try to integrate it into your commercial product. (After running a period of time, it might pause 5 seconds. That is not a bug; I added it in case of anybody trying to integrate it for commercial usage without my permission)
I keep all the copyrights of EchoSight-Pro, although I did it based on interests.

If you have bugs to report or any specific needs from EchoSight-Pro, you can contact this email echosight_pro@126.com

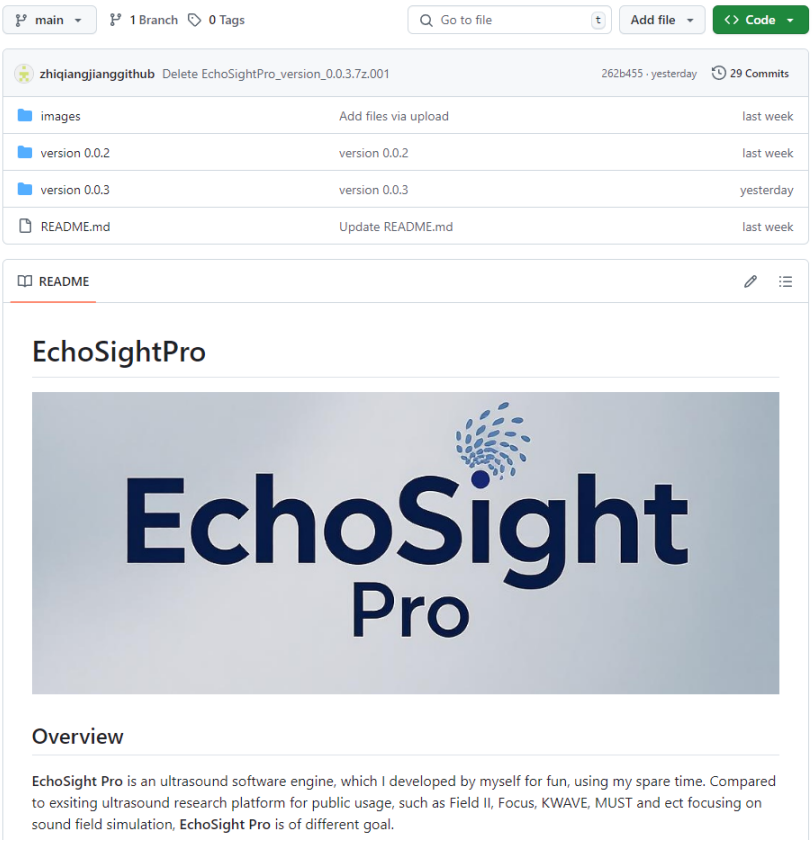
Please feel no shame about that, coz I do not often check email neither.

1. Quick Start

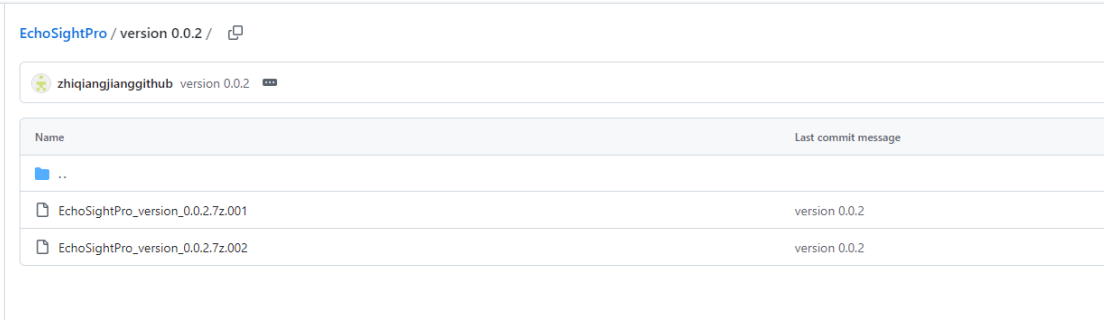
1.1 Download and Run

You can down load the software package from Github or CSDN. Both of them provide latest version download.

GitHub page for download
https://github.com/zhiqiangjianggithub/EchoSightPro



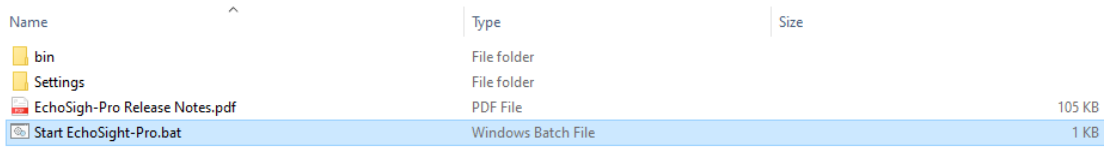
On Github, all history versions are provided. Please download the latest version, current document might not match with old version software.
Due to Github file upload size limitation, no bigger than 30 MB single file is required. The full package might be separated into several smaller files.



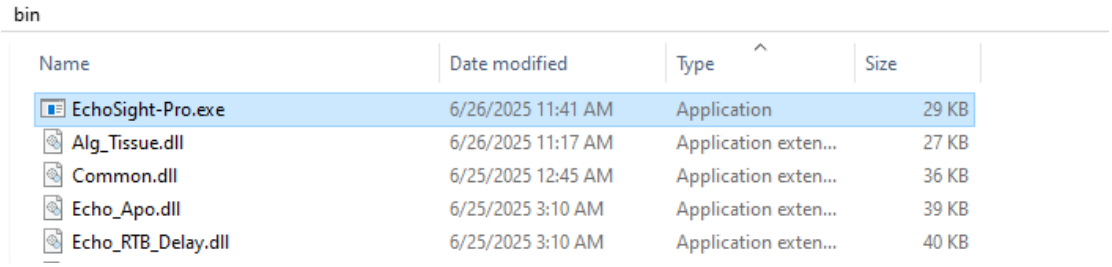
You need to download all the files of the same package, before you unzip it.

After download the package, please unzip it to the folder whatever you like. It's pure green software, no installation needed.

Double click **Start EchoSight-Pro.bat** to start EchoSight-Pro program.



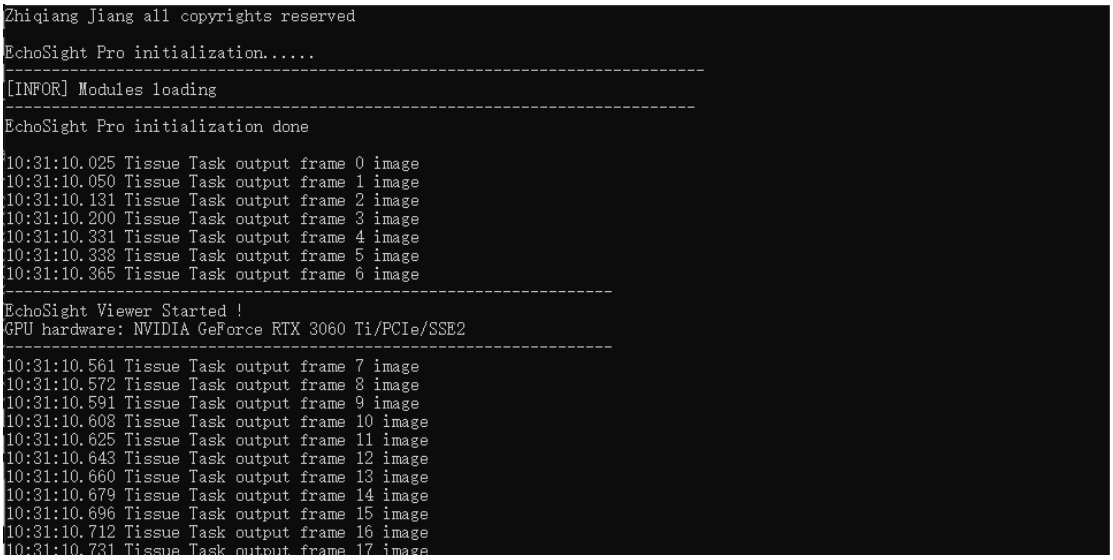
You can also double click **EchoSight-Pro.exe** in bin folder to run the program



1.2 Basic Usage Tricks

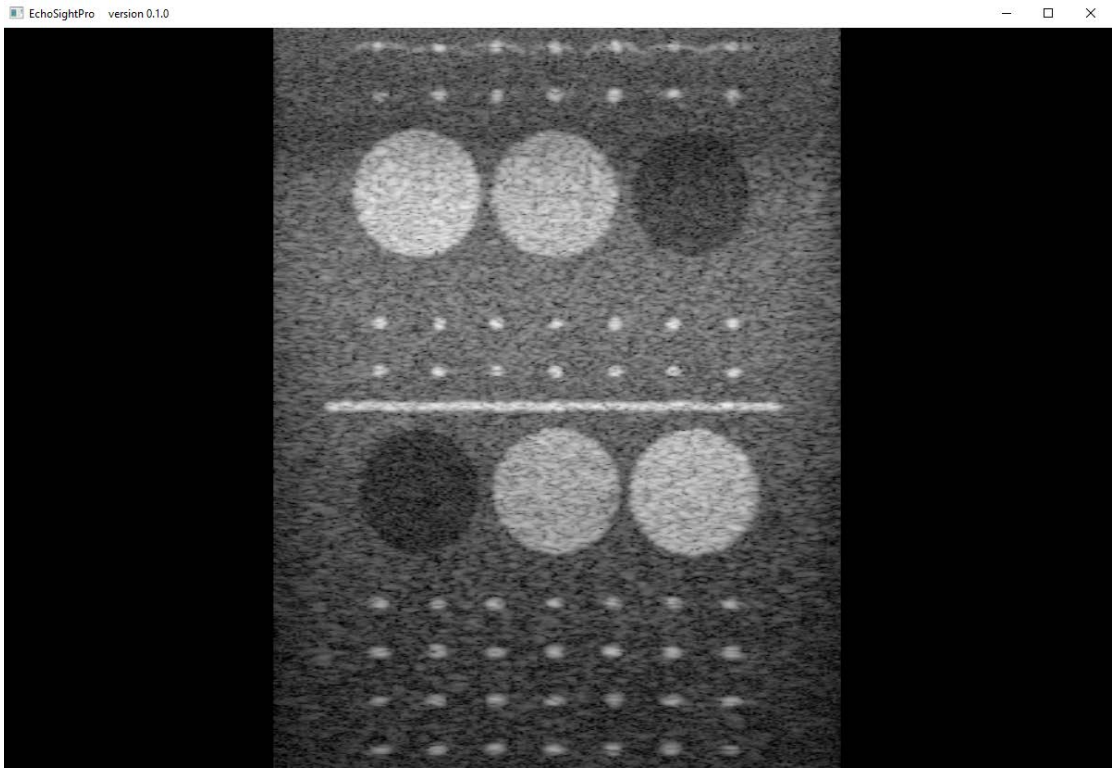
When EchoSight-Pro is running, it will use the full computation power based on default settings. It will pop out two windows, one is command line window, the other is a GUI window (EchoSight Viewer).

Command Line Window



Command line window will give the actual EchoSight-Pro processing status information, including what it is doing right now, as well as the actual speed it is working.

GUI Window



GUI window displays the final results after the whole processing. **Notice**, **the GUI window update rate is not the real processing speed**, you can check the real processing speed at background command line window with accurate time tags.

Important notice, when EchoSight-Pro is running the CPU or GPU is quite busy, if you do not want to burn it out (I'm joking).

Pause the EchoSight-Pro

Use your mouse to left click in the command line window, the whole EchoSight-Pro processing chain will pause and wait.

Resume/Continue EchoSight-Pro

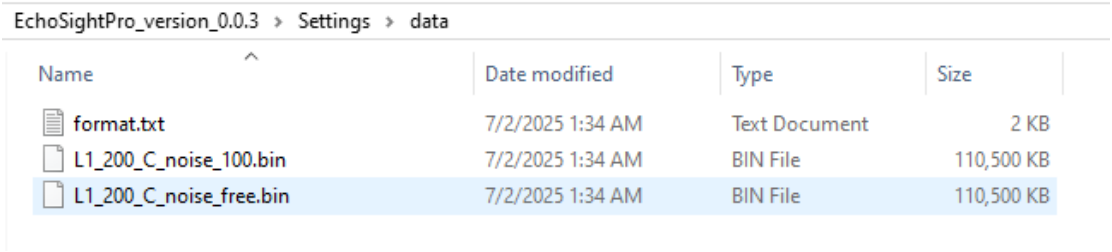
Press enter key in command line window, the EchoSight-Pro will resume from the point it paused at.

2. Data and Configuration

2.1 Raw Data File

2.1.1 file location and format

The EchoSight-Pro uses **raw channel IQ data** as its input. EchoSight-Pro will load in the raw data file from folder **EchoSightPro\Settings\data\XXX.bin**

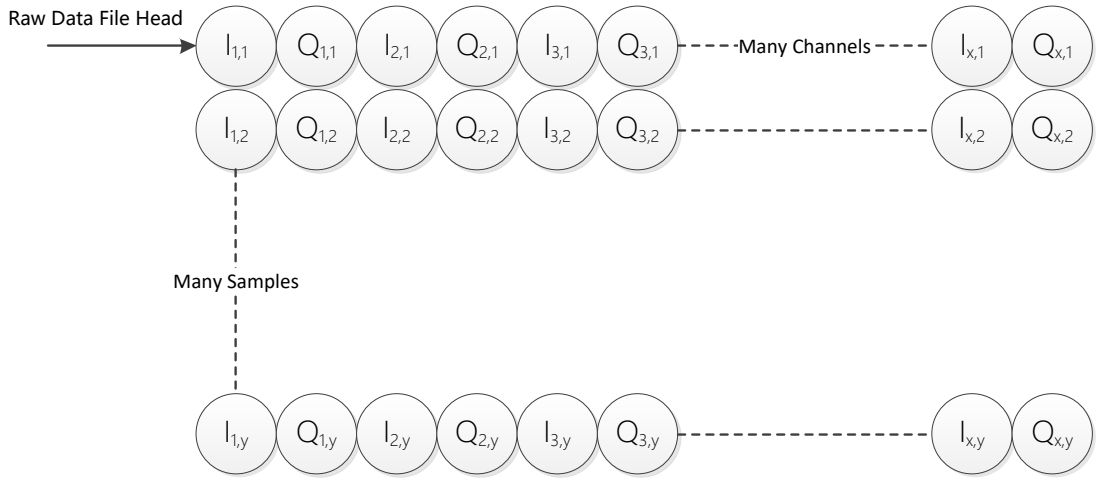


The raw channel data is too huge, I only put 2 basic data in the folder, you can simulate your own channel data based on all kinds of existing ultrasound simulation tools, or capture the real ultrasound channel data on your prototype machine and stores them in file.

To make EchoSight-Pro accept the correct data, the raw data format should follow the format as below:

1. **The data file is stored in raw binary format (not text!!!)**
2. **The data file is stored in the following order**

Format diagram



3. **The data sample is in 16 bits signed integer for each I, Q samples**

You can also know more details by checking the “format.txt” file in the data folder.

PS: Z.Jiang explanation for the data format
The EchoSight-Pro data file storing format is to make the FPGA comfortable after AFE fixed demodulation, and 16 bits integer is the majority AFE default sampling output (in majority cases, it might be 12 bits for 50MHz or 14 bits for 40MHz in reality, except some special chips)

2.1.2 get the raw data file

I have put a basic sample dataset file in the folder within the software package already. You can use it directly. But actually, you should obtain the data, specify the data to your specific needs.

1. Simulation data

There are many well-known tools for this purpose, such as Field 2, Focus, KWave, Must and etc. Their download link listed below
-Field II https://field-ii.dk/
-Focus https://www.egr.msu.edu/~fultras-web/
-MUST https://www.biomecardio.com/MUST/
-KWave http://www.k-wave.org/index.php

After you obtained the data, please reformat the data file in the format described in section 2.1.1.

2. Real Captured data

To use real data is cooler, right? You can use real data as well, EchoSight-Pro do not differentiate them at all.
Some commercial products provide raw data capture before beam forming stage, including but not limited to Versonics, Ultrasonix and some other cooperations**.**

After you captured the data, don't forget the step of transfer the data format as section 2.1.1 requires as well.

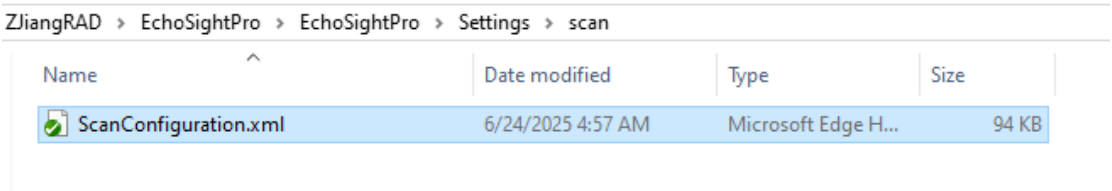
3. Data file-based VS Online Scan

No difference for EchoSight-Pro, the computation, data transfer work load are extremely similar. The file data transfer module is repeating copying the source memory data and doing the transfer instead of using the same data pointer without doing transferring.

2.2 Raw Data Settings

2.2.1 Raw Data Setting file

After you obtain the data already, you left one step ahead to make it work. Set the scan and hardware setting for EchoSight-Pro.
You can find the setting file from folder EchoSightPro\Settings\scan\



Open the XML setting file named **ScanConfiguration.xml** to define your personal needs (Strongly suggest you use NOTEPAD++ to edit the file)

It should look as below

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ROOT>
<ACTIVE_SCAN>LINEAR_STD_1</ACTIVE_SCAN>


<!--linear probe with 1 frame scan, no steering, depth at 4.5CM, focus at 2.5CM, 200 valid scan line -->
<SCAN NAME="LINEAR_STD_1">

<!--linear probe with 3 frame scan, steer at -7/0/7 degree, depth at 4.5CM, focus at 2.5CM, 200 valid scan line -->
<SCAN NAME="LINEAR_STD_3">

</ROOT>
```

2.2.2 Setting file item details

There might be one or more scans listed in the file, ignore them.
Focus on the **ACTIVE_SCAN** item in the root node. It describes which scan setting EchoSight-Pro will use.

Step 1: Create Your Scan
You should create your own SCAN settings. You can start with by copy paste the whole LINEAR_STD_1 context and rename it to your preferred name, let say “MY_COOL_SCAN”.

Step 2: Active Your Scan
You need to change the ACTIVE_SCAN content to “MY_COOL_SCAN”, otherwise it will still use the default one.

Step 3: Customize Your Hardware Setting
You need define your hardware settings according to the data you captured or simulated. There many items below, but for version 0.1.0 only the following ones are allowed modifications, which I marked blue ball by its left.

```
<SCAN NAME="LINEAR_STD_1">
    <PROBE_TYPE>LINEAR</PROBE_TYPE>
    <PROBE_RADIUS_MM>0</PROBE_RADIUS_MM>
    <PROBE_HALF_ANGLE>0</PROBE_HALF_ANGLE>      <!--unit shou
    <ELE_NUM>128</ELE_NUM>
    <RCV_CH_NUM>128</RCV_CH_NUM>
    <RCV_CH_MIN_NUM>6</RCV_CH_MIN_NUM>
    <PITCH_MM>0.3</PITCH_MM>
    <!--USED FOR CUTOFF AND pos_x_mING REFRACTION UNIT:MM, fo
    <LEN_THICK_MM>0</LEN_THICK_MM>
    <!---Micro second is enough for our simulator, nano secor
    <SCAN_TIME_MICRO_SEC>1000</SCAN_TIME_MICRO_SEC>
    <FULL_PULSE_HDTIME_NS>1366</FULL_PULSE_HDTIME_NS>
    <!--TX PULSE HEADING Time of transmit pulse unit:nano sec
    <TX_PULSE_HDTIME_NS>0</TX_PULSE_HDTIME_NS>
    <DATA_SRC>LOAD_FILE</DATA_SRC>          <!--LOAD_FILE:loa
    <DATA_NOISE_MAG>5</DATA_NOISE_MAG>                <!--
    <!--data file size proximate estimation: file size = 2 *
    <DATA_FILES>L1_200_C.bin</DATA_FILES>   <!-- for multiple
    <!---Uniform steer settings for all. Unit degree,negative
    <SCAN_SEQ_STEER>0</SCAN_SEQ_STEER>      <!-- for multiple
    <UserMode>B Fund</UserMode> <!-- Options:B Fund,B Fund Co
```

EIE_NUM: this is the element number of your probe.
RCV_CH_NUM: this defines your machine channel number, associate with the raw data format.
PITCH_MM: this declares your probe geometry element pitch size, unit in mm.
DATA_FILES: this let EchoSight-Pro to know which raw file to load in.
(for version 0.1.0 only 1 file is allowed)

Step 4: Customize Your Scan Group

Scan group defines more details of your scan, one scan group corresponding to raw data file, in the order of the file names you give in the DATA_FILES. (version 0.1.0 only provide 1 file anyway)

```
    <SCAN_GROUP ID = "1">
        <!---SYN_SHIFT_NUM works in channel synthetic mo
        <SYN_SHIFT_NUM>0</SYN_SHIFT_NUM>
        <MLA_NUM>1</MLA_NUM>
        <SYN_NUM>1</SYN_NUM>
        <!--HAN for hanning window, HAM for hamming wind
        <RX_APO>HAN</RX_APO>
        <RX_FN>0.7</RX_FN>
        <!---TX_TYPE can only be: 1.FOCUSED_WAVE 2.DIVER
        <TX_TYPE>FOCUSED</TX_TYPE>
        <TX_FREQ>7100000</TX_FREQ>
        <!--AFE fixed demodulation frequency-->
        <AFE_FD>7100000</AFE_FD>
        <IQ_RATE>12500000</IQ_RATE>
        <ACQ_DEPTH_MM>55</ACQ_DEPTH_MM>
        <TX_FOCUS_MM>25</TX_FOCUS_MM>
        <SOUND_SPEED>1545</SOUND_SPEED> <!---UNIT: meter
        <TX_NUM>200</TX_NUM>
```

MLA_NUM defines how many parallel lines you want to beam form within beam formation stage;
RX_FN defines the receiving F# value;

AFE_FD: defines the fix demodulation frequency AFE chip does. If you obtain the data from simulation, you should modulate it yourself from RF to IQ and gives the AFE_FD value here.

IQ_RATE: defines your AFE output data sample rate after down sampling, you should give its corresponding value here. If you obtain the data from simulation, you can down sample on

your own and gives the value here.

ACQ_DEPTH_MM: your actual scan beam depth.
TX_FOCUS_MM: your actual transmition focus depth, unit in mm. You can set it within the FOV or out of FOV, or quite far away (let say 999999999) for the plane wave case.

SOUND_SPEED: the sound speed value you prefer for beam formation, it will impact the beam forming output.

TX_NUM: this should match the number of transmit line number.

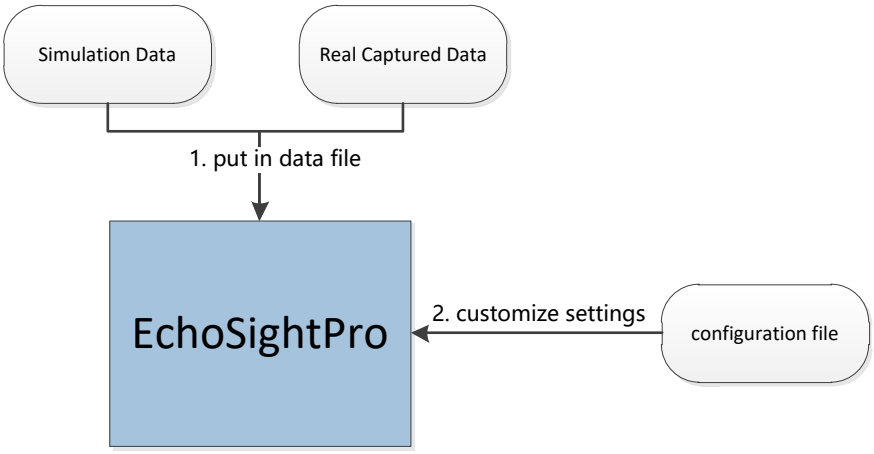
Step 5: Customize Your Scan Sequence
This is the last step.

```
<SCAN_SEQ>
    <!---SCAN INSTRUCTION DEFINITION 3 TYPES: 1.VALID,2.INVALID,3.DUMMY-->
    <TX_ITEM id="0" pos_x_m="-0.01920" steer = "0.0" inversion = "0" mode = "B" frame_id = "1">valid</TX_ITEM>
    <TX_ITEM id="1" pos_x_m="-0.01901" steer = "0.0" inversion = "0" mode = "B" frame_id = "1">valid</TX_ITEM>
    <TX_ITEM id="2" pos_x_m="-0.01881" steer = "0.0" inversion = "0" mode = "B" frame_id = "1">valid</TX_ITEM>
    <TX_ITEM id="3" pos_x_m="-0.01862" steer = "0.0" inversion = "0" mode = "B" frame_id = "1">valid</TX_ITEM>
```

Each tx item defines the property of actual transmit.
Each tx item number should matches the TX_NUM you give above. The pos_x and steer define the transmit kick off position in space. pos_x is in meter unit; steer gives the steering angle in radian unit.

By now, EchoSight-Pro gets everything needed to run.

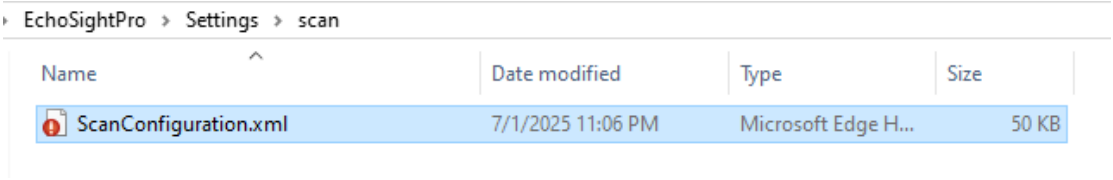
From my perspective, I think it's quite easy.



3. Investigate by Examples

3.1 Kick Off with EchoSight-Pro

Open the “ScanCongiguration.xml” file to confirm the suitable scan set is selected and correct raw data file is chosen.



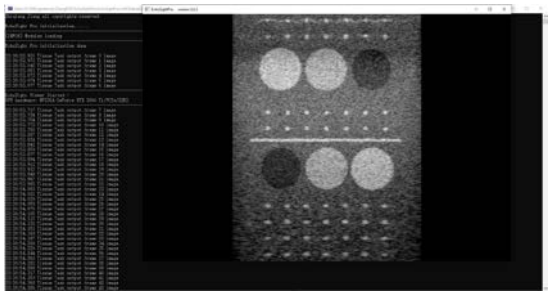
For version 0.1.0, please choose LINEAR_BASIC scan set. Make sure “DATA_SRC” is LOAD_FILE, and corresponding “DATA_FILES” is set as “L1_200_C_noise_free.bin”

```
<ROOT>
<ACTIVE_SCAN>LINEAR_BASIC</ACTIVE_SCAN>


<!--linear probe with 1 frame scan, no steering, depth at 4.5CM, focus at 2.5CM, 200 valid scan line -->
<SCAN NAME="LINEAR_BASIC">
    <PROBE_TYPE>LINEAR</PROBE_TYPE>
    <PROBE_RADIUS_MM>0</PROBE_RADIUS_MM>
    <PROBE_HALF_ANGLE>0</PROBE_HALF_ANGLE>      <!--unit should be radian, data type float-->
    <ELE_NUM>128</ELE_NUM>
    <RCV_CH_NUM>128</RCV_CH_NUM>
    <RCV_CH_MIN_NUM>6</RCV_CH_MIN_NUM>
    <PITCH_MM>0.3</PITCH_MM>
    <!--USED FOR CUTOFF AND pos_x_mING REFRACTION UNIT:MM, for real scan check the probe spec, for simulation l
    <LEN_THICK_MM>0</LEN_THICK_MM>
    <!---Micro second is enough for our simulator, nano second is not reliable on OS either.This will control s
    <SCAN_TIME_MICRO_SEC>1000</SCAN_TIME_MICRO_SEC>
    <FULL_PULSE_HDTIME_NS>1366</FULL_PULSE_HDTIME_NS>
    <!--TX PULSE HEADING Time of transmit pulse unit:nano second, this works for the HW and simulation cases bo
    <TX_PULSE_HDTIME_NS>0</TX_PULSE_HDTIME_NS>
    <DATA_SRC>LOAD_FILE</DATA_SRC>          <!--LOAD_FILE:load from file for simulator,GEN_DATA:no file needed,
    <DATA_NOISE_MAG>5</DATA_NOISE_MAG>                  <!-- Random noise level range numerical value generated
    <!--data file size proximate estimation: file size = 2 * acquisition depth / sound speed * IQ_RATE * REV CH
    <DATA_FILES>L1_200_C_noise_free.bin</DATA_FILES>    <!-- for multiple inputs, separate by ";"-->
```

L1_200_C_noise_free.bin and L1_200_C_noise_100.bin are two files, I simulated for EchoSight-Pro version 0.1.0 usage, and stored in EchoSightPro\Settings\data folder.

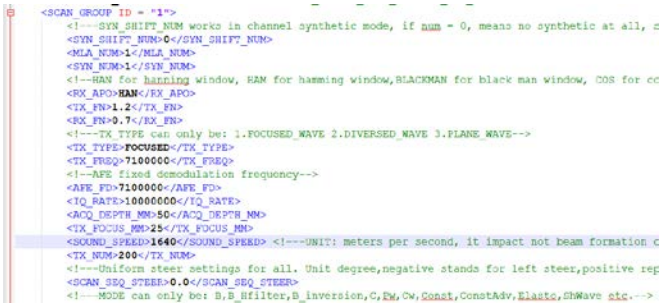
After all the confirmation, click “Start EchoSight-Pro.bat”
You should see the output below



3.2 Modify the Settings

You can change the settings as you like, the beam formed image quality will be impacted as well. Open the “ScanCongiguration.xml” file again, modify the “**SOUND_SPEED**” item under

“SCAN_GROUP” in “LINEAR_BASIC” scan settings.



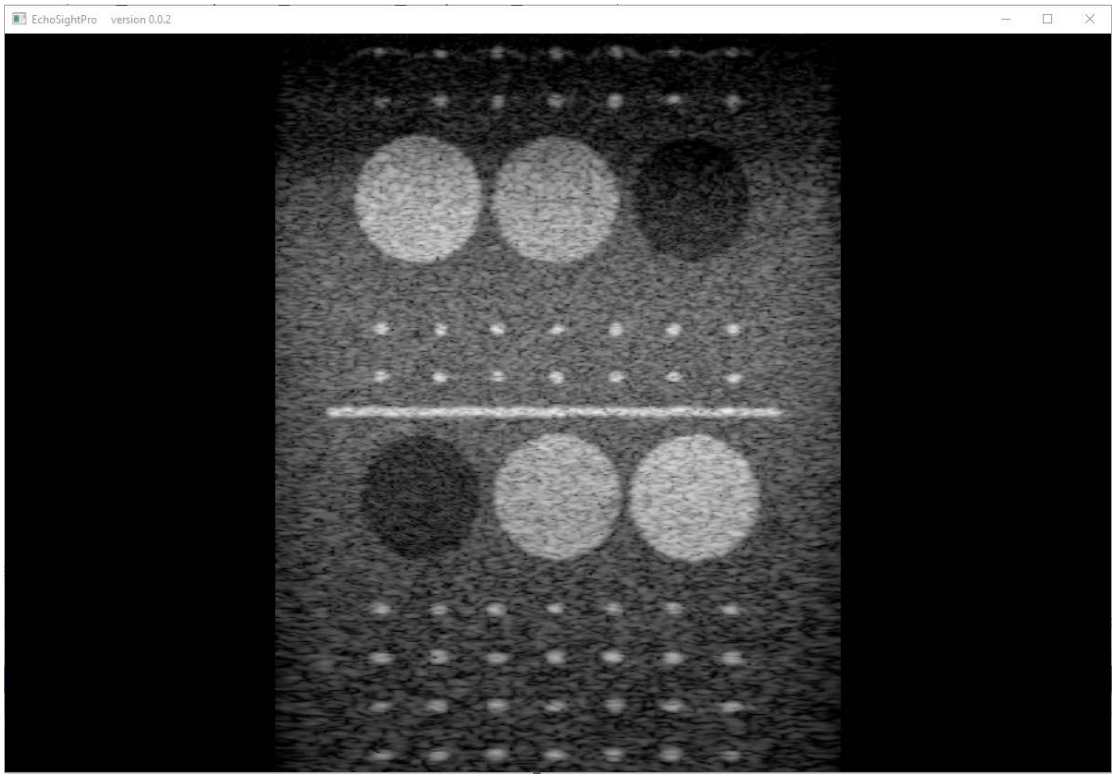
Then double click “Start EchoSight-Pro.bat” to EchoSight-Pro to see what will happen. You can also change other items such as “**RX_FN**”, “**RX_APO**” to see what will happen.

3.3 Have MLA and RTB

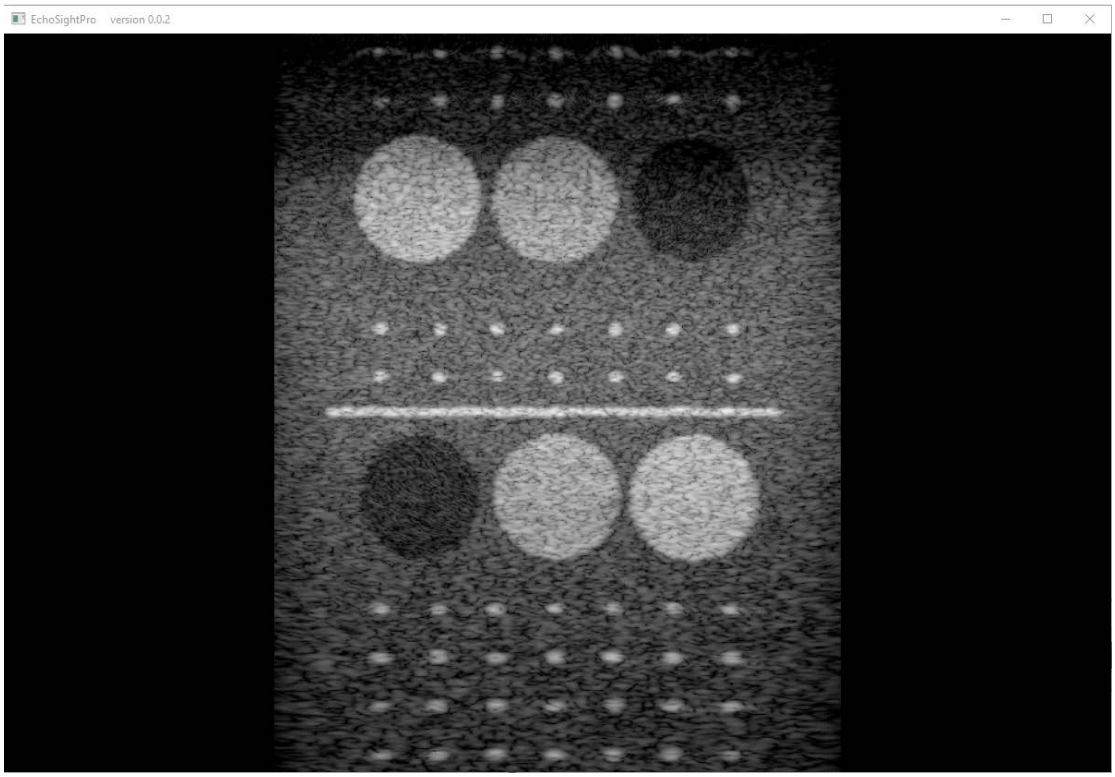
MLA is one of most important parameters in settings. It orders EchoSight-Pro to beam form multiple lines with 1 fire channel data.

```
<SCAN_GROUP ID = "1">
    <!---SYN_SHIFT_NUM works in channel synthetic mode, if num = 0, means no synthetic at all, sec
    <SYN_SHIFT_NUM>0</SYN_SHIFT_NUM>
    <MLA_NUM>5</MLA_NUM>
    <SYN_NUM>1</SYN_NUM>
    <!--HAN for hanning window, HAM for hamming window,BLACKMAN for black man window, COS for cosi
    <RX_APO>HAN</RX_APO>
```

When your MLA is bigger than 1, the RTB will enable automatically as well. **Both MLA and RTB are build in EchoSight-Pro already and work in real time**. Try it to see what will happen. Two Images are shown below, both of them are of 200 transmissions, but the first image is of 1 MLA while the second image is of MLA 4, which is 4 times line density along horizontal direction. Check the difference below



Transmit 200 times, with MLA 1



Transmit 200 times, with MLA 4

Look at how much horizontal resolution we have obtained by higher MLA!
RTB attempts to fix the sound wave front position error by shifting A/D sample position after receiving the signal. Although it cannot completely fix that, it does fix majority of errors in wave front position mis-match.
Turn on MLA will increase the EchoSight-Pro processing working load obviously.
Let say your data rate is 5 Giga Bytes per second, MLA 20 means EchoSight-Pro needs to process 5 Giga Bytes data calculation 20 times per second, which means 100 Giga Bytes of data computation per second on your computer.
It's quite a lot data, baby. That won't be a small challenge for your computer and Z.Jiang's system engineering design capability as well.

3.4 Synthetic Now

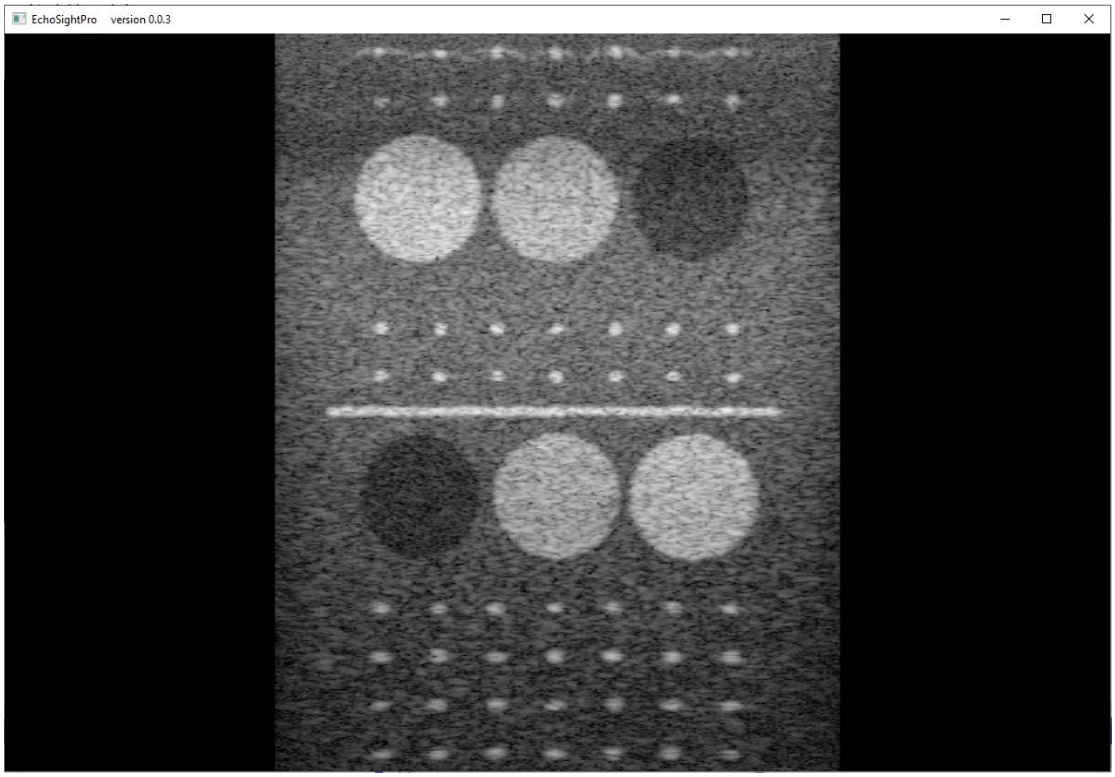
Synthetic aperture is a traditional technique widely used in radar, sonar, ultrasound system design. **EchoSight-Pro has built in real time synthetic technique already**. But remember to turn on the multiple line acquisition beforehand, otherwise we do NOT have multiple data to synthetic image at all.

```
<MLA_NUM>8</MLA_NUM>
<SYN_NUM>8</SYN_NUM>
```

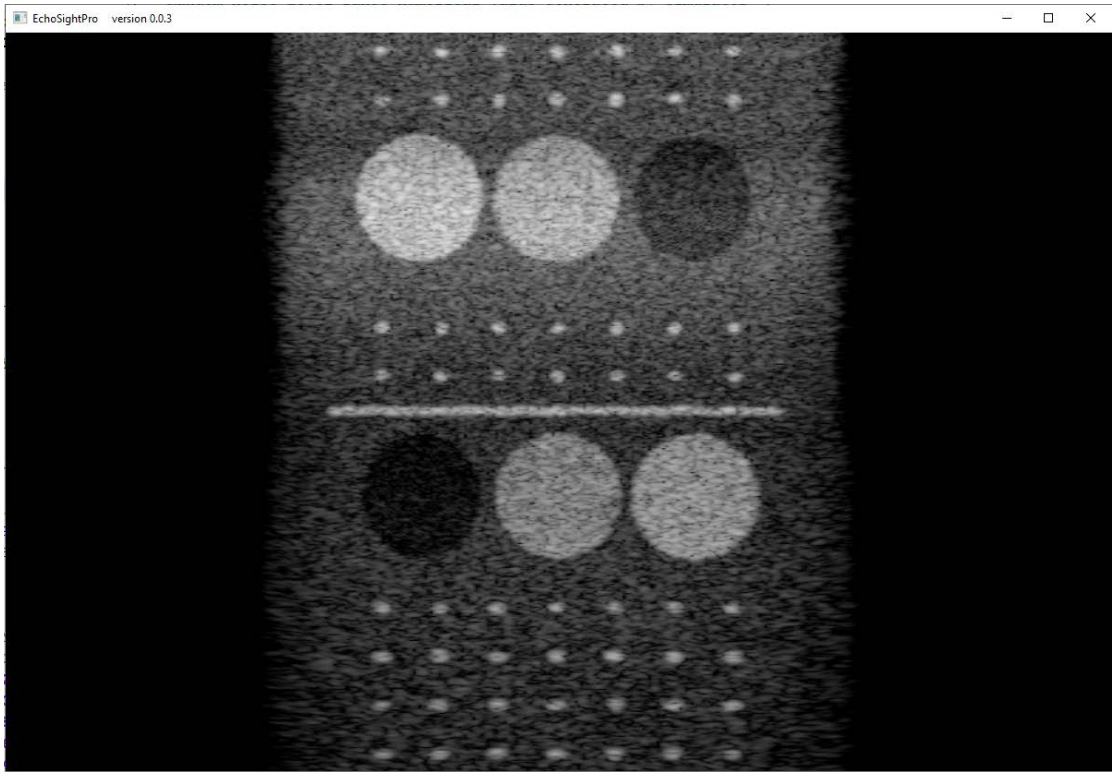
Check the two images generated by EchoSight-Pro below, which generated from the identical channel data where I added in value 100 random noises in the channel data. (please use

L1_200_C_noise_100.bin instead of **L1_200_C_noise_free.bin**)

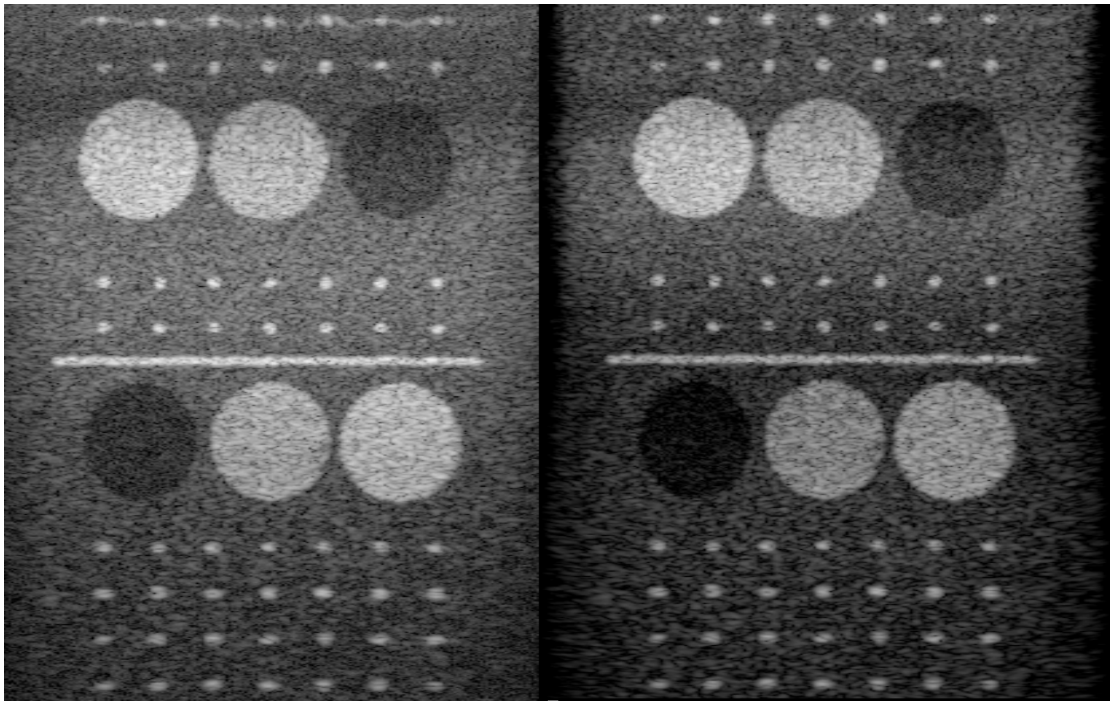
The first image is of MLA 1, SYN 1(which means no synthetic at all)



Check the second image for comparison, which is of MLA 24 SYN 24.



I put them side by side below, it will be more obvious



The IQ improvement is dramatic, mainly on 3 facets.

1. Point becomes tighter, grating lobe drops.

This is mainly caused by synthetic, and also benefits from RTB (EchoSight-Pro build in RTB and works automatically) some.

2. The penetration (SNR) improves at far fields a lot.

This is also caused by synthetic, SNR will increase 3dB every time when you double your synthetic number. This makes signal pops out while noise drops down, which leads to signal penetration improvement.

3. Contrast resolution increases a lot.

Contrast resolution is an important IQ factor, which was introduced by Legendary ultrasound company ACCUSON. It is used to evaluate the machine's capability to differentiate different region.

Synthetic combination of signal will lead to overall signal drop a little, but darker area drops more, which leads to the difference between the white and dark area increase in return. This makes machine's contrast resolution increase. (Please check the cyst differentiation between the background)

4.5 Steer some Angles

TBD

4.6 RealTime Noise On

4.7 Multiple Frames

TBD

4.8 Check EchoSight-Pro Limits

TBD

5. Performance Check

The performance of version 0.1.0 can **vary depends on your computer hardware** capabilities.
My testing is based on my personal computer

CPU	Memory	GPU
Intel i5-12400	16G @ 2400MHz	Nvidia 3060Ti 8GB

Testing summary

EchoSight-Pro v0.1.0 CPU Only Testing Summary			
MLA	**SYN**	**Data Transfer Rate (GB/Sec)**	**Data Process Rate (GB/Sec)**
1	1	6.8	6.8
4	4	3.1	12.5
8	8	1.7	13.6
16	16	1.0	16.0
24	24	0.58	13.9

v0.1.0 is working on pure CPU mode only, I am still quite happy about its performance.
When EchoSight-Pro has its GPU mode supported in next version (like old EchoSight did), it will be quite interesting.

6. Author foreshows

EchoSight-Pro is completely new project based on new design concept. There almost no

connection between EchoSight and EchoSight-Pro. The only connection is both of them are from the same author, that's me.

Version 0.1.0 is still at early stage. Compared to EchoSight, it is quite far away on functionality facets. But with new architecture design, EchoSight-Pro will surpass EchoSight on all facets later, and can provide some super cool capabilities beyond imagination.