AUDIO FINGERPRINTING

PRACTICAL EXPERIMENT

NOUR SAFFAF - [15/MAR/2018]

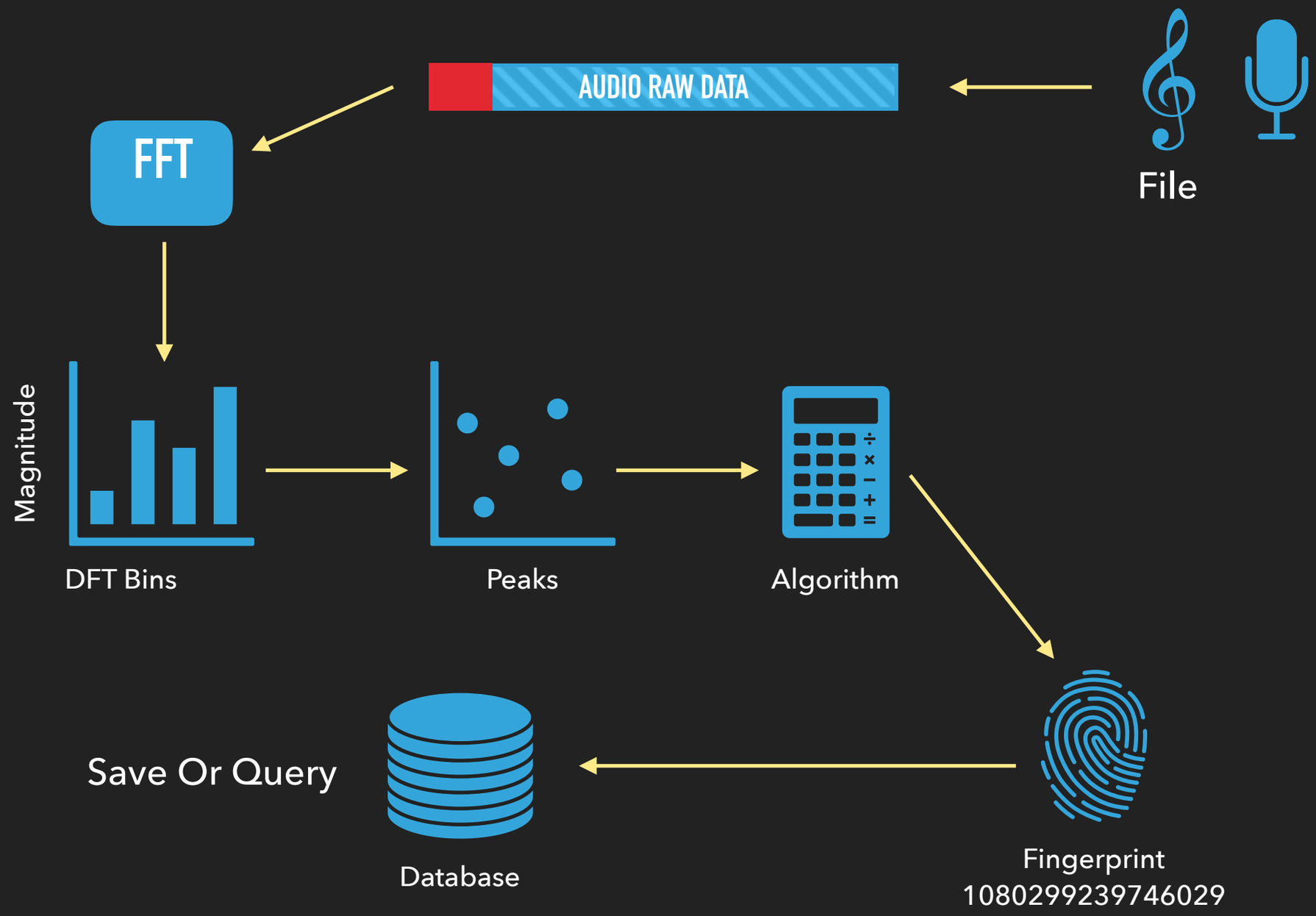
CONTENT

- Theory Quick Sketch
- First Experiment
- Stage 1: Extract Raw Audio Data
- Stage 2: Fast Fourier Transform
- Stage 2 Challenges
- Stage 3 - Frequency Bands
- Stage 4 - Fingerprinting
- Stage 5 - Database
- Frontend & Backend
- Server-less Development In Mobile
- Experiment Results
- Business Perspective

INTRODUCTION

THE JOURNEY BEGINS

THEORY QUICK SKETCH



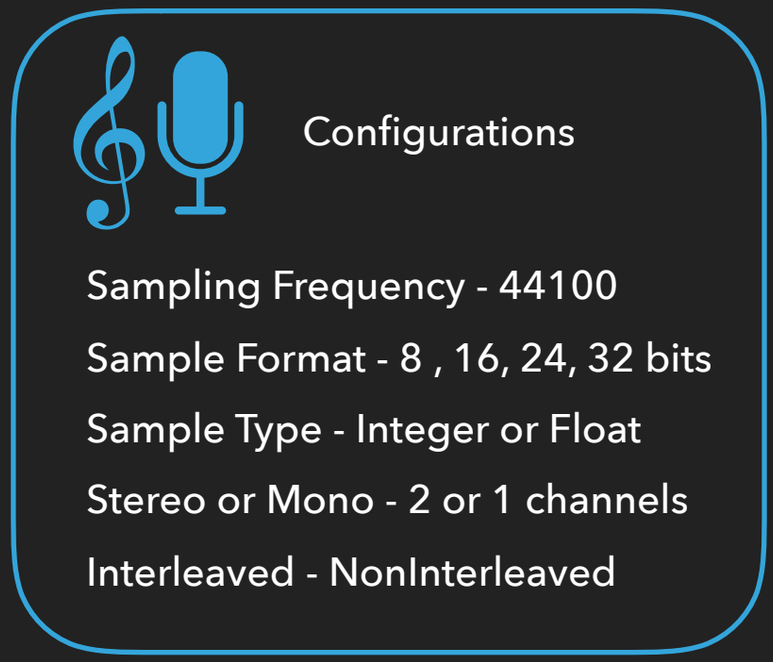
FIRST EXPERIMENT

- Followed theory, developed a sample software in 10 days
- Generated > 10,000 fingerprints for a ~3.5 minutes song
- Started listening for 10s, 20s, 30s … etc
- Results = ZERO fingerprints matched

- Why the experiment has failed?!!
- What to present now?

STAGE 1 – EXTRACT RAW AUDIO DATA

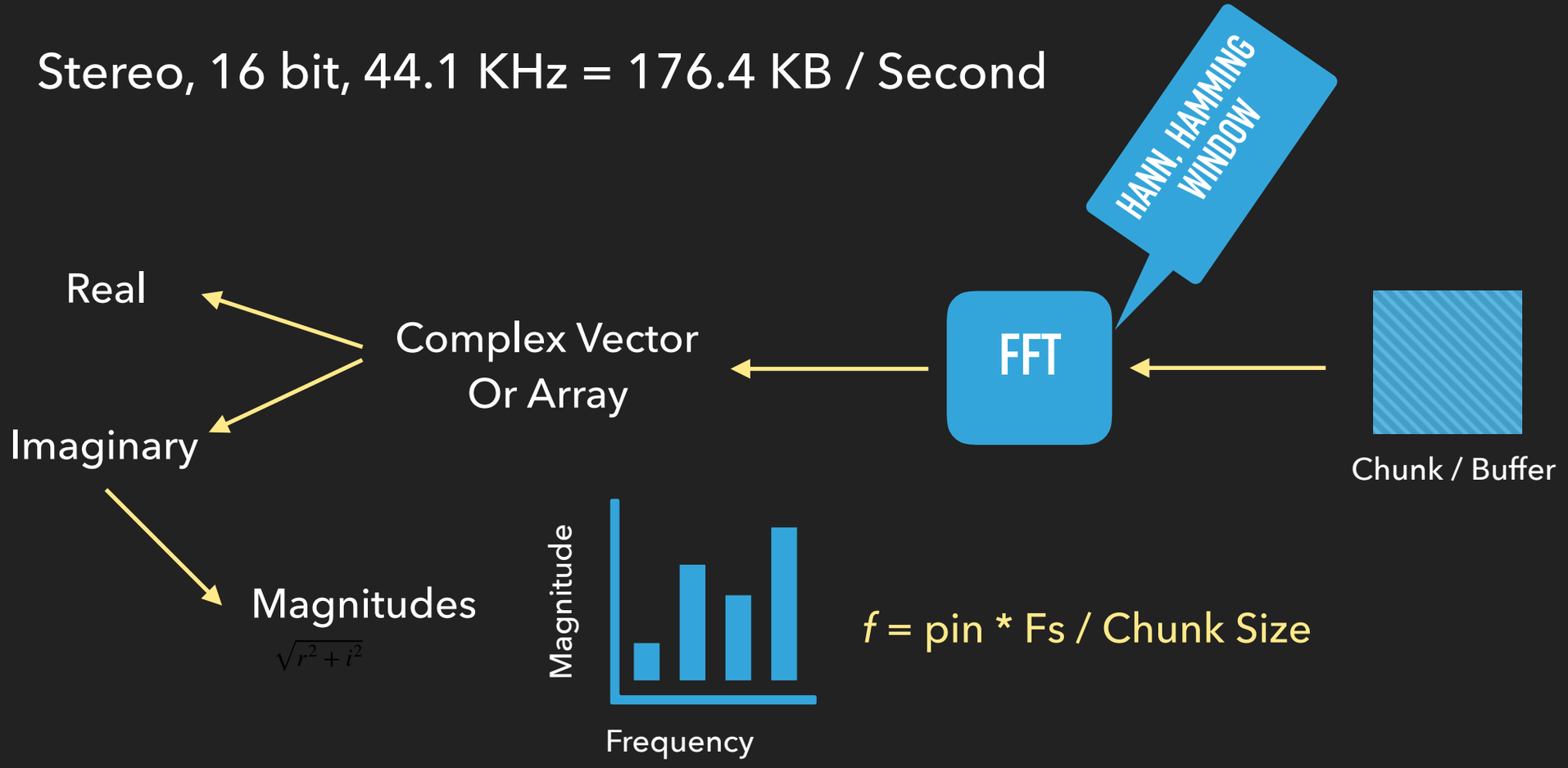
- Don’t work on compressed audio files (mp3)
- Audio file and Mic recording configurations must match!
- What configurations should I select?
- Should I test or trust the system?



- Tip: work on wav audio files

STAGE 2 – FAST FOURIER TRANSFORM

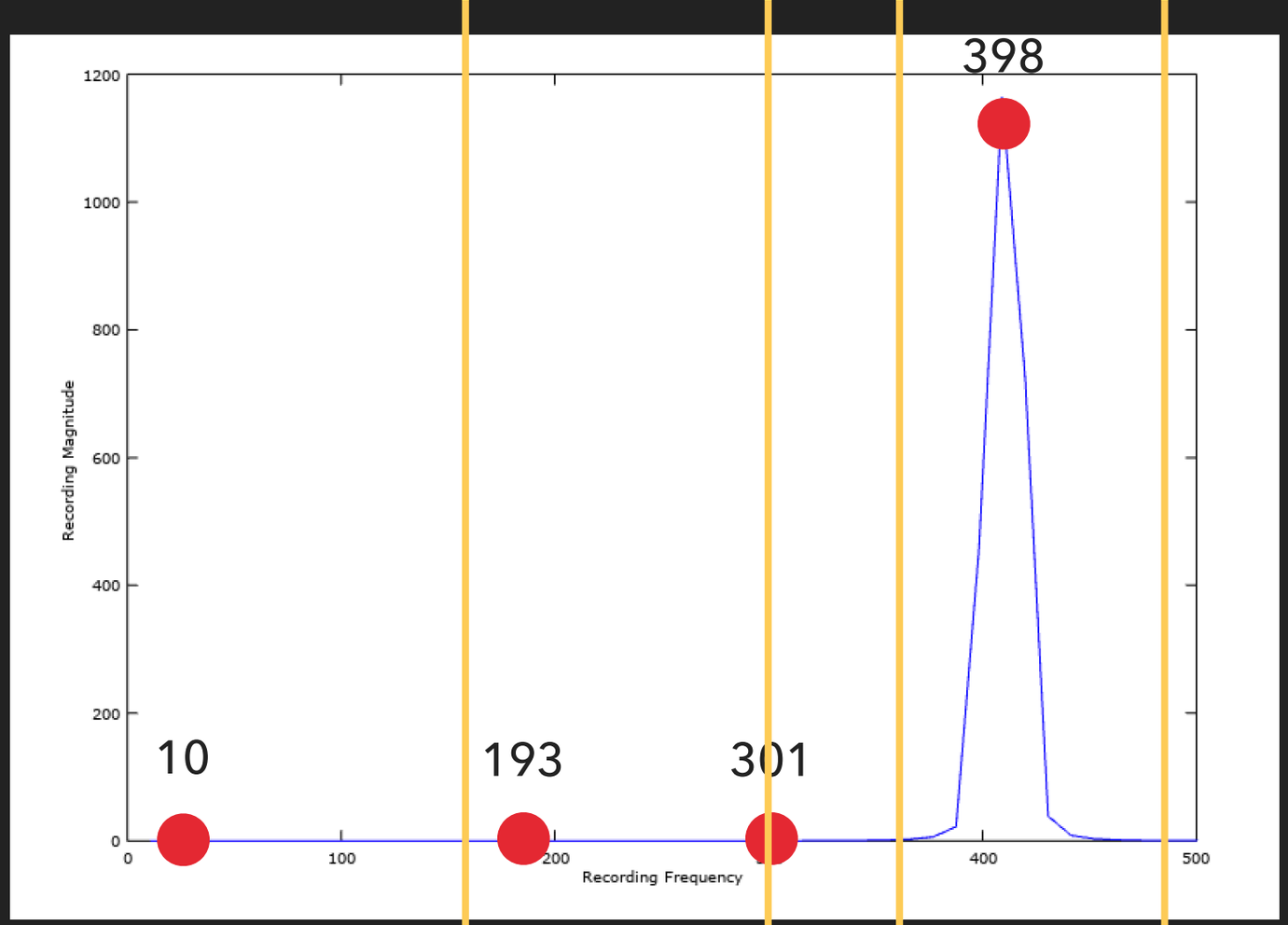
- Only one chunk at a time [4, 8, 16 , 32 …] kb
- How many chunks per second?
- Mono, 16 bit, 44.1 KHz = 88.2 KB / Second
- Stereo, 16 bit, 44.1 KHz = 176.4 KB / Second



STAGE 2 – CHALLENGE

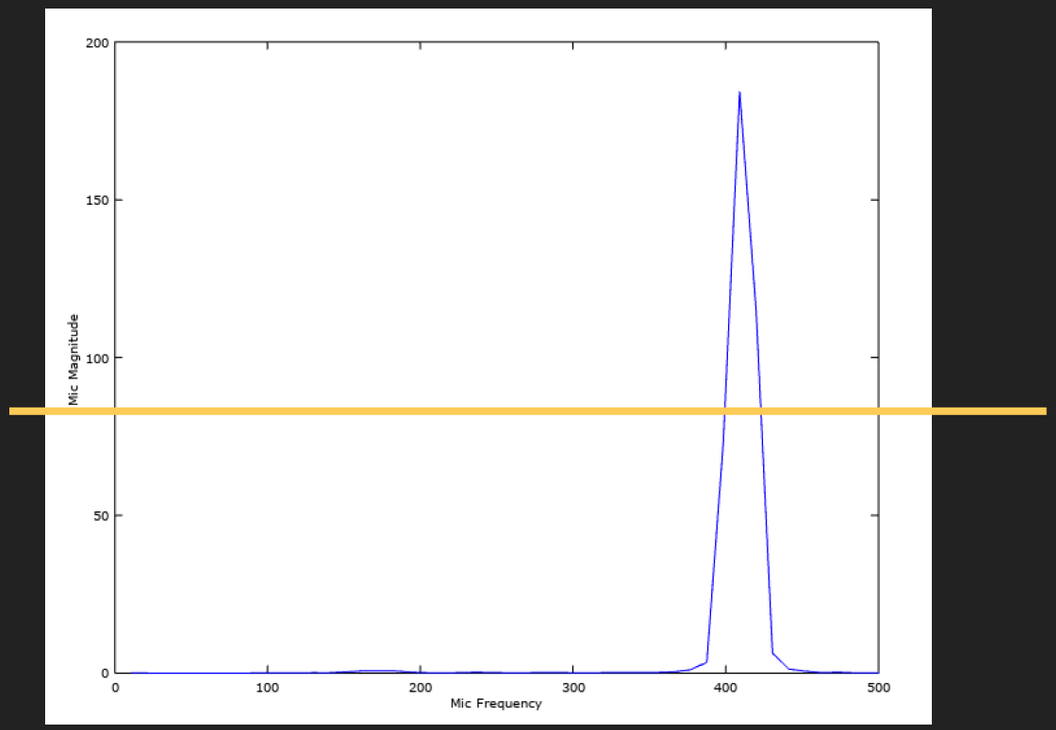
- Is the output of FFT on the audio file and mic recording are the same?
- Use Visualisation!! Matlab/Octave
- Test single pure tone (for example 400 Hz)

[10, 193, 301, 398]
for all chunks



STAGE 2 – MIC CHALLENGE

- Recording peak [10, 193, 301, 398]
- Is the Mic producing similar peak?

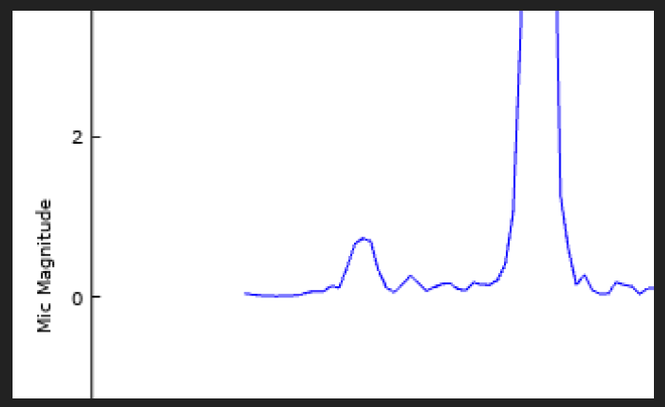


[139, 161, 301, 398]

[150, 161, 226, 398]

[129, 161, 215, 398]

[150, 193, 236, 398]

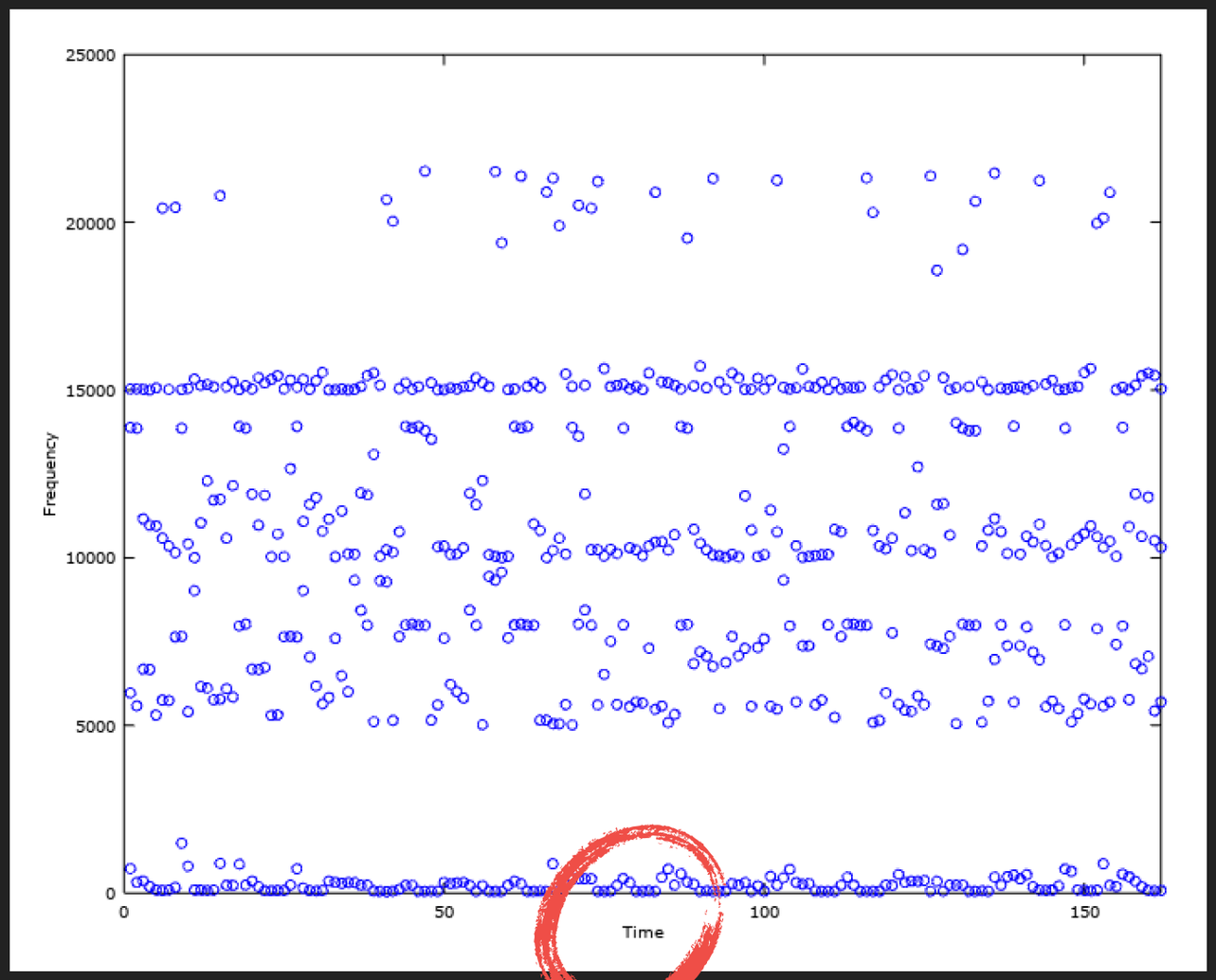


Normalisation/Silence Threshold

[0,0,0,1]

STAGE 3 – FREQUENCY BANDS

- What frequency bands should I choose?
- Need help? Create the Peaks Map



Frequency Range

20 Hz to 20 kHz

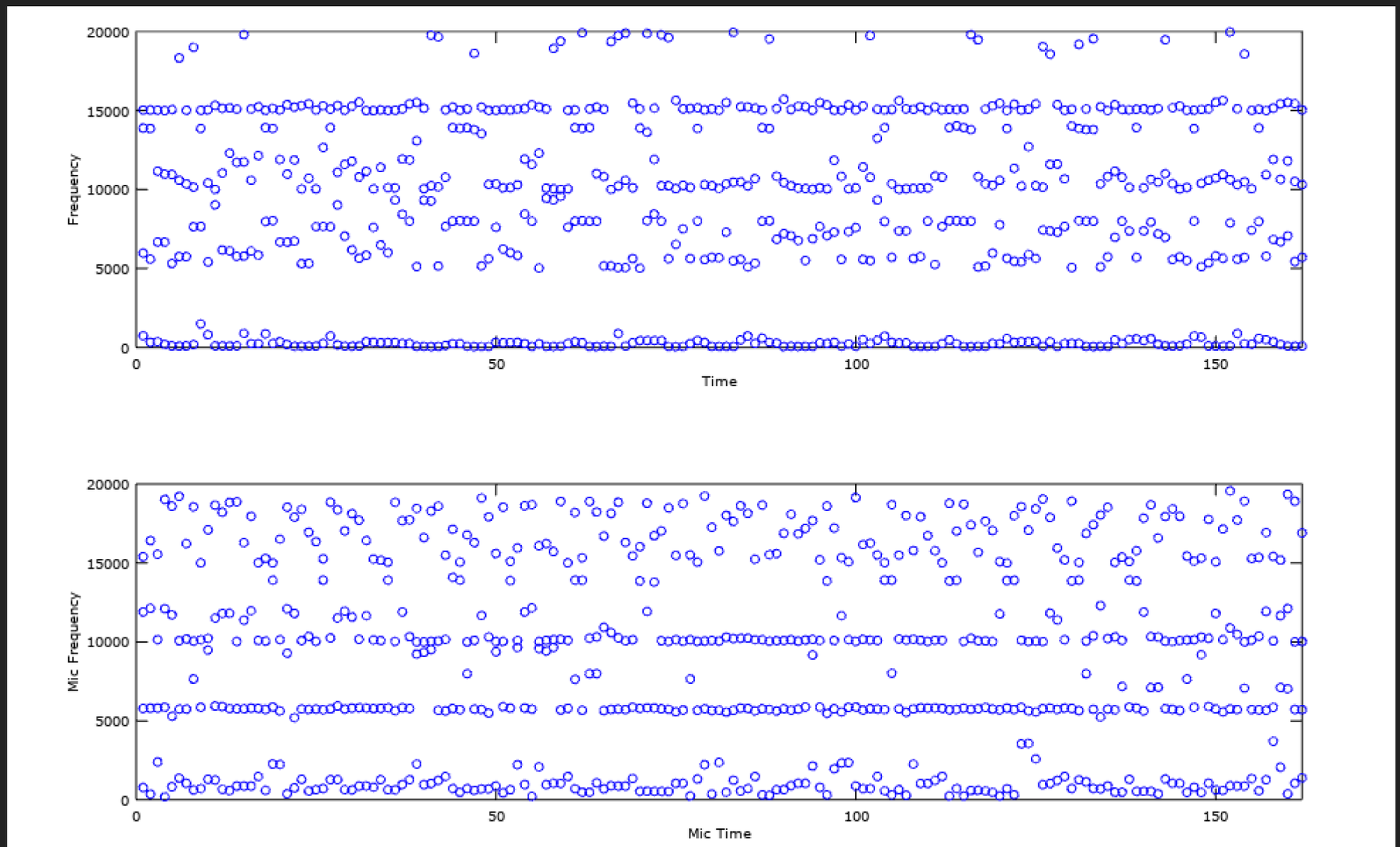
C4 = 261.6 Hz

300 Hz to 3.4 kHz

Compare To Your Mic

Your will get different results

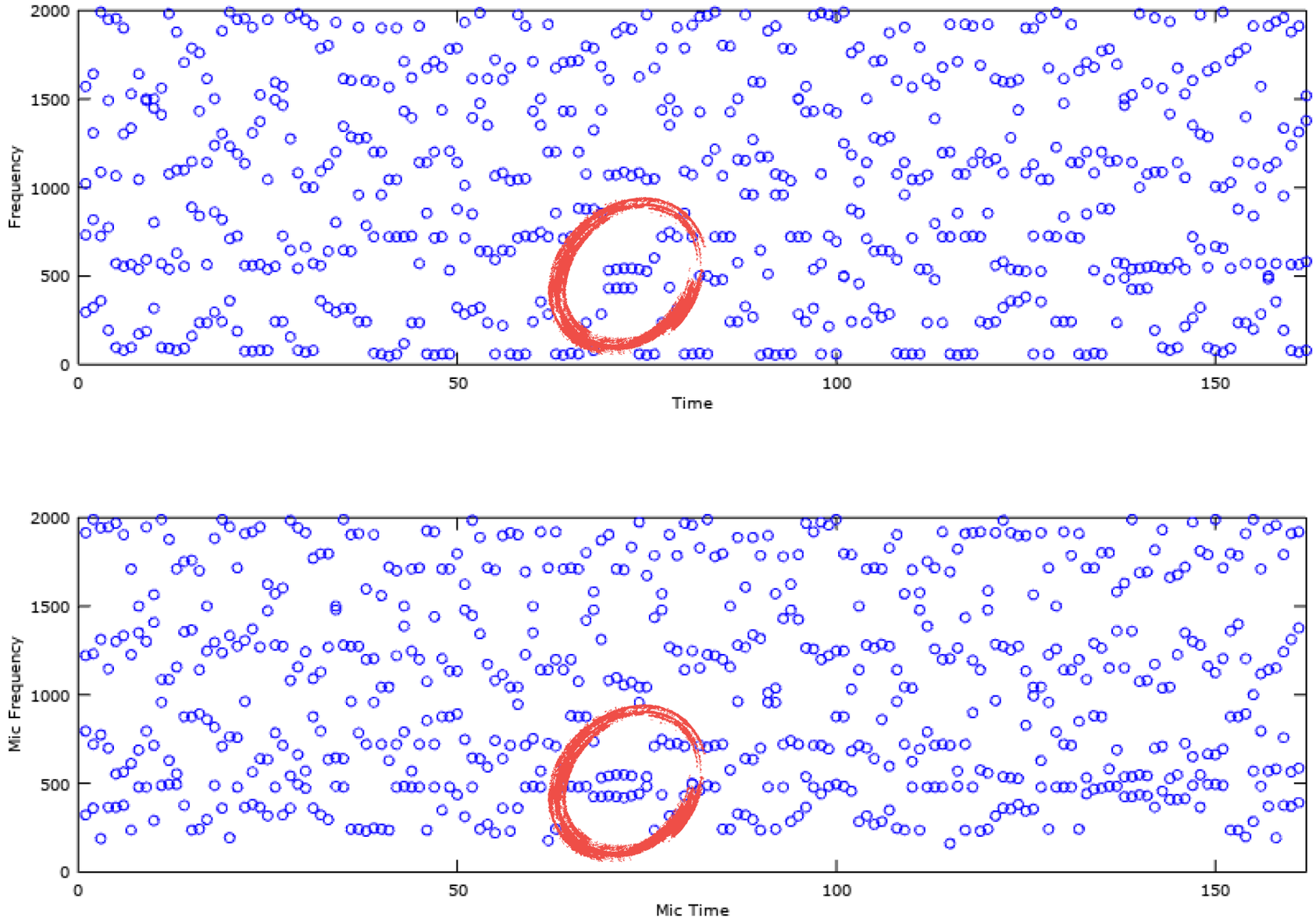
STAGE 3 – AUDIO VS MIC PEAKS MAP – 5K BANDS



Bands

~5k Hz

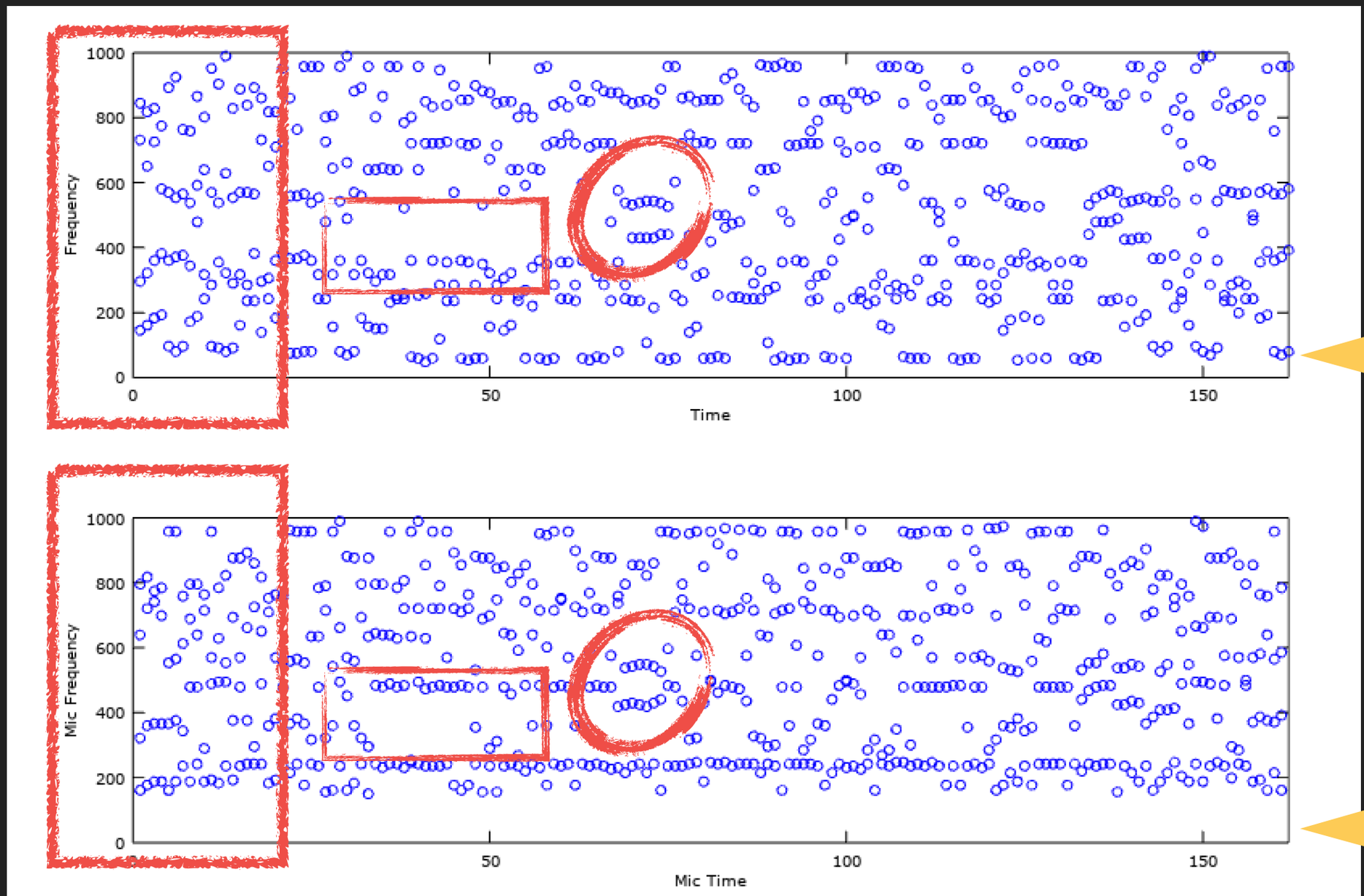
STAGE 3 – AUDIO VS MIC PEAKS MAP – 500 BANDS



Bands

~500 Hz

STAGE 3 – AUDIO VS MIC PEAKS MAP – 250 BANDS



Bands

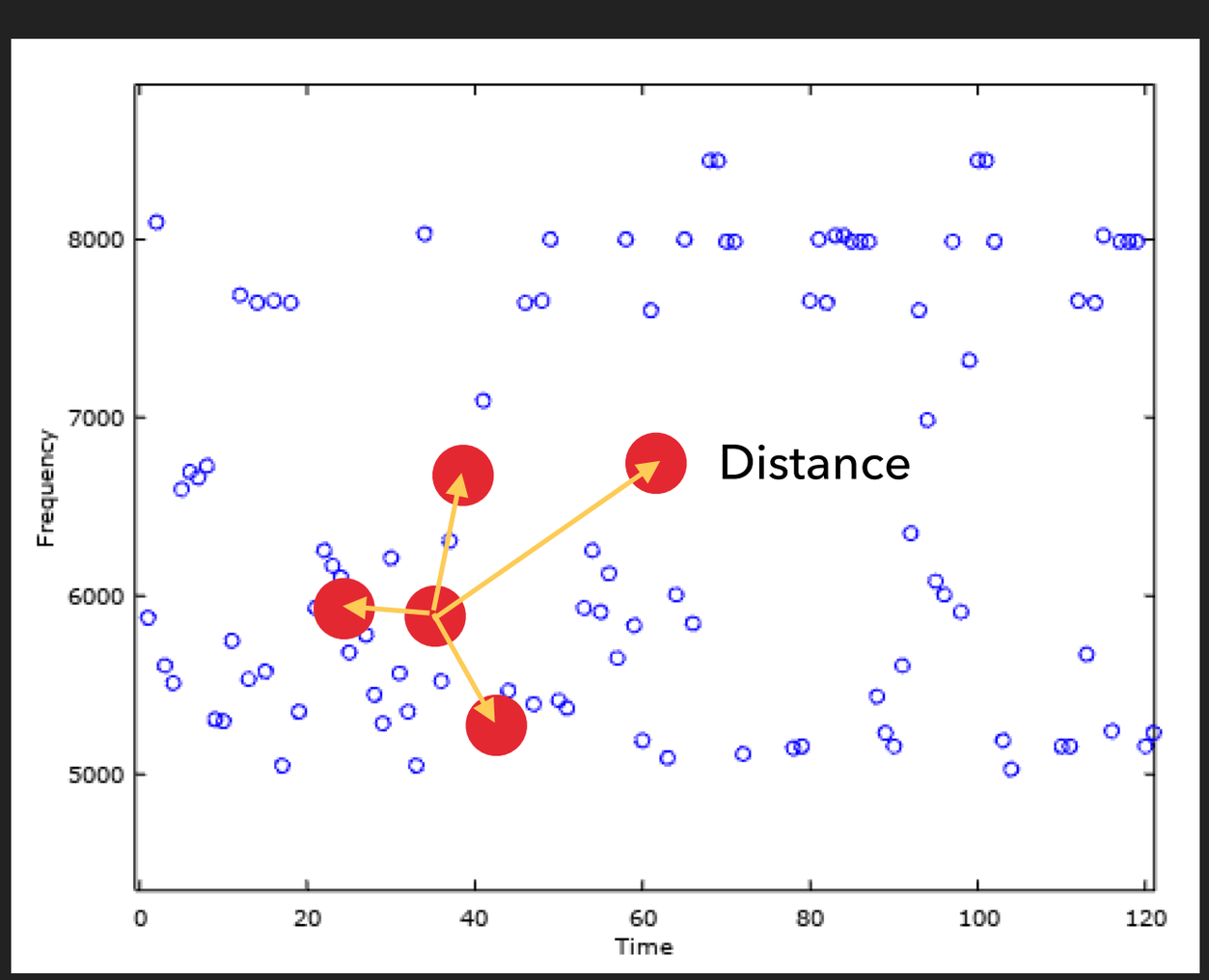
~250 Hz

STAGE 3 – FREQUENCY BANDS SELECTION

- From the visualisations we conclude:
- Narrowing the frequency bands results in more peak matches
- Narrowing the frequency bands may result in higher duplicate peaks (smaller DFT bins)
- Narrow frequency band is needed for Mic recording to capture peaks at specific low frequencies, for example 170 Hz
- In my opinion, better detection and less duplicates could be achieved using dynamic frequency bands based on the chunk magnitude's average
- The choice of the fingerprinting algorithm may influence the bands selection
- For this experiment, I selected fixed 4 frequency bands between 20 Hz and 1k Hz

STAGE 4 – FINGERPRINTING

- Now we selected the bands, so what next?
- Algorithm? Technique?
- Uniqueness or Duplicates?

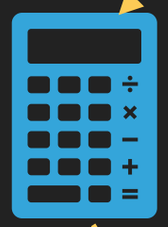


Information

Point 1 Frequency 5800 Hz

Point 2 Frequency 6700 Hz

Distance 60 - 35 = 25 ms



Algorithm

Hashing Technique

10902398493209094380

Fingerprint

STAGE 5 – DATABASE

- Unique fingerprints or Duplicates?

FingerPrints FPs	Time	Song Id
10902398493209094380	0.2s	Song_1
20802367543209094534	1.3s	Song_2
68780279824398492094	19.4s	Song_100

Unique FingerPrints Table

SHA-1
Will Drevo

377 MB

5.4 million FPs

45 songs

FingerPrints FPs	Songs Id
10902398493209094380	Song_1, Song_10
20802367543209094534	Song_2, Song_13
68780279824398492094	Song_7

Duplicates FingerPrints Table

My Experiment

18.9 MB

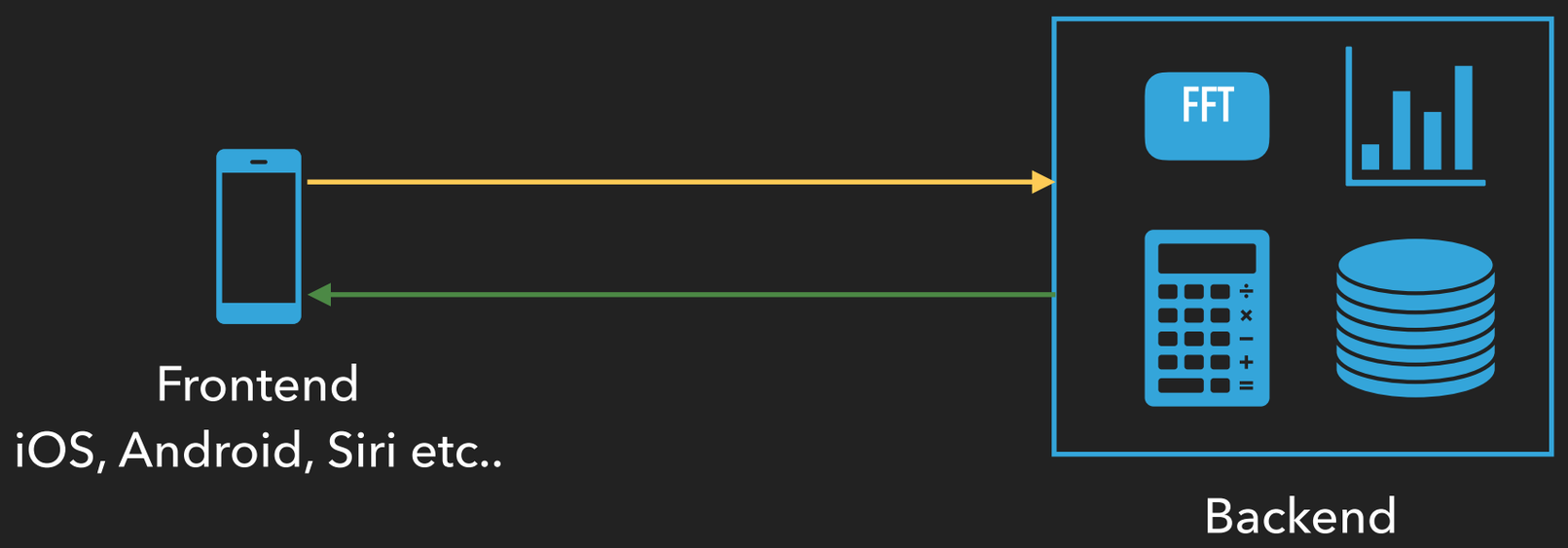
64699 FPs

15 songs

size is not linear

FRONTEND & BACKEND

- Fingerprinting is a pre-processing step
- Mobile app needs only to send the chunks/buffers to the server
- Server computes the FFT, selects peaks, etc…
- Server queries the database - finds best match
- Server sends the result back to the app



SERVER-LESS DEVELOPMENT IN MOBILE

#	iOS	Android
1	Core Audio	InputStream, AudioRecord
2	vDSP	JTransforms, TarsosDSP … etc
3	Swift , Obj C	Java, Kotlin
4	Peak as a Struct	Peak as a Class
5	Core Data, realm	SQLite, Room, realm

EXPERIMENT RESULTS 97.3%

8 KB Chunk	30s	15s	10s	5s	3s
Song 1	✅	✅	✅	✅	✅
Song 2	✅	✅	✅	✅	✅
Song 3	✅	✅	✅	❌	❌
Song 4	✅	✅	✅	✅	✅
Song 5	✅	✅	✅	✅	✅
Song 6	✅	✅	✅	✅	✅
Song 7	✅	✅	✅	✅	✅
Song 8	✅	✅	✅	✅	✅
Song 9	✅	✅	✅	✅	✅
Song 10	✅	✅	✅	✅	✅
Song 11	✅	✅	✅	✅	✅
Song 12	✅	✅	✅	✅	✅
Song 13	✅	✅	✅	✅	✅
Song 14	✅	✅	✅	✅	✅
Song 15	✅	✅	✅	✅	✅

Failed first time due to noise, succeeded three times consecutively

Success with close score for next match

BUSINESS PERSPECTIVE

- Congratulations you have created Shazam-Like App!
- You are going to be a rich man  $54m revenue 2016 (The Verge)
- Not really….. 

- 1st problem: Millions of songs, commercials, TV shows etc..
- 2nd problem: Patents everywhere.. Even Shazam was sued in 2009 🤔
- 3rd problem: Competitors (Apple, SoundHound, ACR Cloud)
- 4th problem: Marketing budget ($100 K)?
- Another problem: Why customers would migrate to your app?

REFERENCES & GOOD ARTICLES

- http://willdrevo.com/fingerprinting-and-audio-recognition-with-python/
- http://coding-geek.com/how-shazam-works/
- https://labrosa.ee.columbia.edu/matlab/fingerprint/
- https://www.theverge.com/2017/12/11/16761984/apple-shazam-acquisition

CONCLUSION

THE JOURNEY ENDS

FOR NOW!

ONE MORE THING – SAWTTI APP

- 1-Rag'n'Bone Man - I am Human
- 2-Ed Sheeran - Shape of you
- 3-Adele - Rolling in the deep
- 4-Mark Ronson - Uptown Funk
- 5-Earth, Wind & Fire - September
- 6- PSY - Gangnam Style
- 7- Sia - Cheap Thrills
- 8- Ariana Grande - Side To Side
- 9- The Chainsmokers - Closer
- 10- Shakira - Waka Waka
- 11- Lou Bega - Mambo No. 5
- 12- Luis Fonsi - Despacito
- 13- Major Lazer & DJ Snake - Lean On
- 14- Beyoncé - Naughty Girl
- 15- Los del Rio - Macarena

Q&A