Senior Project

Presentation

Visuals can be faster than words.

“Presenter shows ‘lyrics player’ to audience”

“Audience gasps in awe”

“Presenter moves on to next slide…”

Steps to desire this thing.

- Enjoy listening to music
- Listen to foreign music
- Decide you want subtitles

subtitles for lyrics and with translations in realtime.

So I realized a night ago while writing this presentation.

There exists multiple better solutions to what I created to solve, like no way… again! 😅

Here are the best and worse cases of solving a problem for a programmer.

Best case: code does it all ‘magic’

Worst case: USER enters the realm of the equation.

Programmers hate when USER has to get involved to solve anything.

Seriously, we can automate everything if given enough time to create a fully automated solution.

Fairy tale ending hmm what is it going to be?

Best End: USER listening to music and benefits from subtitle service. (Seeing both native users language and foreign while song plays is believed benefit)

Worst End: USER gets the best end but has to wait on translations for subtitles because our system isn't working fast enough to get translation done because we rely on that ugh… USER!

TIME measures the amount of sequences of events required to complete a task. The smaller the TIME to complete the task the better and when no time needed we have magic!

TIME shares the answer to the universe and magic… 😈

In other words…

Les contes de fées sont faits pour être défaits…

The fairy tales are made to be defeated…

So I’m transitioning on…

Moving my development time in favor of a new solution which popped into my head.

Proposed solution will decrease time to get close to magic level.

The tool I built will compliment the new way of doing a particular task by existing to modify output from the new solution.

New proposal see next slide…

Solution came to me when making these slides.

- If we get USER to sing song as album is played.

- Singer can be in a controlled environment, where **camera aimed at mouth by laptop webcam** or **whatever**,… could be setup to produce accurate time positioning of native lyrics by a 💋 lip reading solution when **USER sings song as it is played**.

- Accurate translations of the text from lip reading solution is done by sending native sung lyrics to an Google API Service for translation.

- Holy shit, we just need USER to sing a song mimicking an artist singing the original album songs for us.
  (Only need sung once by USER)

Approaching magic

- Almost little to no time is needed from USER.
- A random USER must exist whom can sing just like artist (pitch is not necessary). USER just needs to **mimic** start and stop of words sung which are picked up by **lip reading technology**.
- Lip reading technology does exist but I didn’t think about it till now. 🙁

See this repository: github.com/sagioto/LipReading as an example of lip reading technology

Anyway that is the plan
for the future!

Used these languages

- Javascript
- C++
- Objective-c
- Swift

With a 2013 MacBook Pro Retina

Google Cloud Platform
ubuntu
JetBRAINS
Develop with pleasure!
WS
github
SOCIAL CODING
CL
node
JS
express
AC
socket.io
nw.js
npm
mongoDB
>_

Youtube

https://www.youtube.com/user/jsconfeu/videos

Y Hacker News

https://news.ycombinator.com

the holy &#@! how did i end up down this path… 😊

- I started creating a C++ project, which would be the process gathering music information.

- After writing a decent amount of code, I realized on a mac it would have to be Objective-c code to retrieve music information from iTunes. (**Discovered it’s possible to have Objective-c inside C++ code**)

- Once I had a C++ project getting music information by calling Objective-c code. I built the GUI for the Lyrics Builder in a html page. Eventually I realized it would make more sense to have the html page bundled in NW.js instead of just hosting the html page.

- Somewhere in time I ran into memory leaking issues and Swift had been released to I migrated all the code to Swift. I still had issues so I had to have Swift and Objective-c code.

So where does the rabbit hole go?

fin.