

**ANTEPROYECTO DEL PROYECTO FIN DE GRADO**

DEPARTAMENTO: Ingeniería Audiovisual y Comunciones

TITULACIÓN: Grado en Ingeniería de Sonido e Imagen

APELLIDOS ALUMNO: Crespo García

NOMBRE ALUMNO: Marcos

DNI: 51010864S correo: m.crespog@alumnos.upm.es

TEMA: Diseño de aplicaciones de audio sobre Raspberry-Pi

TÍTULO PREVISTO: Implementación de un corrector de tono (*Pitch Corrector*) en tiempo real para voz sobre *Raspberry-Pi*

TÍTULO PREVISTO EN INGLÉS (obligatorio): Real time pitch corrector implementation for voice over Raspberry-Pi

FECHA PREVISTA EXAMEN: Julio 2020

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| La memoria se va a redactar en idioma: | ESPAÑOL | **x** | INGLÉS |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| El proyecto se va a defender en idioma: | ESPAÑOL | **x** | INGLÉS |  |

PROPUESTA DE MIEMBROS DEL TRIBUNAL:

TUTOR: ……………………………………………………………………………………….

SECRETARIO: ………………………………………………… V.ºB.º: ……….…

PRESIDENTE:

Si PFG externo:

DIRECTOR: ……………………………………………………………………………………

Empresa/Organismo: ………………………………………………………………………..

Financiación: Dpto.: …….. Escuela: …….Otros: ………………………………………….

DESCRIPCIÓN DEL PFG (longitud mínima dos páginas)

1. INTRODUCCIÓN

Desde mediados del siglo XX se han usado distintos métodos para corregir el tono de diferentes señales musicales, con no demasiado éxito. No fue hasta 1997 cuando Antares Tecnologies lanzó al mercado Auto-tune, un plugin disponible por entonces para Pro Tools, el DAW más importante del momento. Este fue inventado por Andy Hildebrand, el cual se dedicaba a mapear datos sísmicos con la finalidad de encontrar petróleo. Hildebrand cambió el crudo por los estudios musicales cuando vio que los algoritmos utilizados también servían para trabajar con señales de audio.

Los técnicos de los estudios musicales se sorprendieron de la facilidad y rapidez con la que se podía corregir el tono de las voces y que no estuvieran desafinadas. Tras esto en 1998 la artista y cantante Cher lanzó la primera grabación comercial con este efecto, llamada “*Believe*”. Este evento dio el pistoletazo de salida a la popularización de este plugin cuya forma de uso ha ido variando con el paso de los años hasta hoy, donde casi la totalidad de música comercial lo utiliza.

Este uso (y abuso dirían algunos) masivo del efecto de audio ha llevado a que estos artistas también quieran usarlo en sus actuaciones en directo. Para este cometido distintas empresas han lanzado módulos hardware y software capaces de procesar y trabajar con señales de voz en tiempo real, permitiendo afinar la voz a una tonalidad y escala concreta con latencias muy bajas. El mayor inconveniente de estos módulos es su alto precio y la opacidad que existe en torno a su funcionamiento.

Con este proyecto se pretende desarrollar e implementar un corrector de tono de bajo coste capaz de funcionar en tiempo real.

1. OBJETIVOS.

El objetivo que se quiere conseguir es una implementación de bajo costo de un corrector de tono o *pitch corrector*. Este se desarrollará para su funcionamiento sobre el ordenador de placa única *Raspberry-Pi* y a una tarjeta de audio *Audio Injector*.

Para el desarrollo e implementación del proyecto se valorará el uso de distintos algoritmos, según trabajen en el dominio del tiempo o de la frecuencia, eligiendo aquellos que favorezcan el rendimiento para su uso en tiempo real.

Todo el diseño del sistema se hará con la premisa de que el sistema funcionará con señal de voz.

1. ESPECIFICACIONES Y RESTRICCIONES DE DISEÑO

Dada la limitada potencia que posee el mini-ordenador se utilizarán algoritmos y parámetros dentro del procesado de señales que se adapten a trabajar con señales de voz (señal limitada en frecuencia y monofónica).

Se trabajará la voz introducida en un único canal como señal mono.

Se valorará el uso de algoritmos que trabajan en tiempo o en frecuencia según la potencia disponible en la *Raspberry-Pi*. Un mayor tiempo de procesado supone mayor latencia, la cual si supera (((Xms))) dificulta el uso en tiempo real.

La corrección de voz se aplicará como mínimo sobre la escala musical Do Mayor con *La4* a 440Hz.

1. METODOLOGÍA DE TRABAJO PROPUESTA

El software se desarrollará sobre un ordenador personal con Matlab. Este se compilará y volcará sobre *Raspberry-Pi* a través de Simulink y los múltiples módulos y *add-ons* que posee para trabajar sobre el mini-ordenador, conectado a través de un cable Ethernet, manejando la consola de este a través de SSH con el software Putty.

La metodología de trabajo será *Git Flow.* Esta consiste en trabajar en distintas ramas, atomizando el trabajo en tareas simples que se van integrando unas con otras hasta conseguir el producto final. De esta manera es fácil organizar en tiempo todo el proceso de desarrollo, permite localizar con mayor facilidad los errores y es accesible en todo momento para su seguimiento en la plataforma *GitHub*.

Durante el desarrollo se irá probando cada uno de los módulos y pasos que conforman la corrección de tono con señales de prueba, vocales y no vocales.

1. DESGLOSE DE TAREAS Y CRONOGRAMA

‘’En el que se planifica las actividades, su secuenciación y su duración con fechas.’’

1. RECURSOS PREVISTOS

Ordenador personal con SO Windows 10

*Raspberry-Pi 3B+*

Tarjeta de audio *Audio Injector*

Matlab 2019a

Simulink

Putty

Señales de tests

Micrófono dinámico

Cableado Ethernet, RCA y XLR

1. PRESUPUESTO (SI ES EL CASO)
2. BIBLIOGRAFÍA

El alumno se compromete a realizar un proyecto original, referenciando en todo caso el origen de la información utilizada, para no incurrir en casos de plagio.

Fdo.:

|  |  |
| --- | --- |
| Fecha:  Tutor:  Fdo.: | Aprobado por la Comisión de Ordenación Académica:  Fecha: |