# -Motivação

Existem diversos algoritmos de reconhecimento de padrões sonoros, dentre eles, alguns focados em reconhecimento de fala, transformação de fala em texto ou até mesmo padrões musicais (Afinação de instrumentos e identificação das notas musicais através do campo de frequência), porém nenhuma destas aplicações permite a manipulação livre dos dados, com a intenção de trata-los de forma geral, visando produzir um resultado condizente com os tipos de dados oferecidos como um espaço amostral na entrada do algoritmo.

Logo, a partir da falta de generalidade nas comparações presentes nos algoritmos atualmente, gostaria de criar um algoritmo que de forma mais ampla possibilite analisar a semelhança desses dados oferecidos e produzir uma margem de confiabilidade ao detectar o mesmo padrão, independentemente do tipo significativo de dados, como por exemplo, o canto de um pássaro ou o ressoar de um violino.

<https://2.bp.blogspot.com/-mhsT0jYklf0/V2yZ-DPafMI/AAAAAAAAA7M/78_dxDNPgyAwYwQzhyYBL_H07z6MVnusQCLcB/s1600/frequencias.jpg>

[1]

<https://abrilexame.files.wordpress.com/2016/09/size_960_16_9_twitter-passaros1.jpg?quality=70&strip=info&resize=680,453>

[2]

<https://upload.wikimedia.org/wikipedia/commons/f/f6/Old_violin.jpg>

[3]

Tendo como base este tipo de pré-coleta de dados, execução do algoritmo, recebimento de uma entrada em forma de áudio e produção da saída de dados relevantes (como por exemplo, o resultado da comparação do áudio) podemos formar a seguir objetivos bem claros para este projeto.

# -Objetivos

1. Possibilitar a formatação de uma amostra de áudio em informações relevantes e bufferizáveis utilizando a Transformada rápida de Fourier (FFT), que converte um sinal do seu domínio original para uma representação no domínio da frequência, obter uma array que descreva a magnitude das frequências representáveis em um bitrate de 44000Hz, ou seja, 1Hz até 22000Hz.

<http://www.multiteide.es/wp-content/uploads/2016/10/Fourier_transform_time_and_frequency_domains_small.gif>

[4]

1. Utilizar a informação da matriz gerada para localizar as frequências em um determinado espaço de tempo, podendo assim obter informações contíguas que me permitam quantificar e caracterizar determinadas características deste áudio para futura comparação.

<http://www.multiteide.es/wp-content/uploads/2016/10/Ejemplo2bis-768x677.jpg>

[5]

1. Armazenar essas informações pré-processadas para formar um banco sólido de dados, aumentando assim a confiabilidade na hora da comparação, diminuindo o número de falsos positivos (obtenção de resultados positivos na hora da comparação que, na verdade, deveriam ser negativos), que ocorrem por alguma flutuação em uma variável de ambiente de onde se obtém a informação a ser comparada, como por exemplo, algum tipo de ruído ou barulho de fundo na gravação do áudio.

<http://www.multiteide.es/wp-content/uploads/2016/10/Ejemplo4-768x767.jpg>

[6]

“Figura mostrando o ruído na amplitude do áudio e como isso afeta a obtenção das frequências”

1. Conseguir utilizar informação tanto armazenada quanto em tempo real, para comparação e resposta imediata.
2. Responder ao usuário do algoritmo qual informação de áudio se assemelha à entrada fornecida e qual o grau de confiabilidade desta resposta.

# -Aplicações e considerações finais

É possível encontrar diversas aplicações com um algoritmo tão abrangente, dentre elas estão: Reconhecimento de fala, Reconhecimento de notas musicais, afinador para diversos instrumentos, Aplicações na área ambiental para estudo de cantos e sons emitidos por diversos animais, Softwares didáticos que ensinam a entonar a voz em aulas de canto, Identificação de pessoas pelo seu tom de fala em aplicações na área de segurança, etc.

A ideia principal deste projeto é o funcionamento do algoritmo com a intenção de ser uma base para a obtenção dos dados, dependendo da implementação e do que será feito com esse dado de forma a dar sentido à comparação, ou seja, o algoritmo pode ser utilizado em diversos projetos maiores e mais especializados, que foram supracitados como exemplo, os quais necessitem dessa obtenção, filtragem e distinção dos dados de forma indiscriminada do que se busca analisar.

# -Bibliografia

FFT:

<https://en.wikipedia.org/wiki/Fast_Fourier_transform>

<https://www.youtube.com/watch?v=rozd8MwQYYs&list=PLX-LrBk6h3wQczbulbyeg7gFr1_T1YN4c&index=1>

<https://www.youtube.com/watch?v=0EQIBSLaXpg&list=PLX-LrBk6h3wQczbulbyeg7gFr1_T1YN4c&index=3>

<https://www.youtube.com/watch?v=0EQIBSLaXpg&list=PLX-LrBk6h3wQczbulbyeg7gFr1_T1YN4c&index=3>

Sons e FFT

<https://docs.google.com/presentation/d/1zzgNu_HbKL2iPkHS8-qhtDV20QfWt9lC3ZwPVZo8Rw0/pub?start=false&loop=false&delayms=3000#slide=id.g5a7a9806e_0_30>

Espectrogramas, vibrações e FFT

<http://www.multiteide.es/2016/10/656/>

Fontes de imagens

[1] <https://2.bp.blogspot.com/-mhsT0jYklf0/V2yZ-DPafMI/AAAAAAAAA7M/78_dxDNPgyAwYwQzhyYBL_H07z6MVnusQCLcB/s1600/frequencias.jpg>

[2] <https://abrilexame.files.wordpress.com/2016/09/size_960_16_9_twitter-passaros1.jpg?quality=70&strip=info&resize=680,453>

[3] <https://upload.wikimedia.org/wikipedia/commons/f/f6/Old_violin.jpg>

[4] <http://www.multiteide.es/wp-content/uploads/2016/10/Fourier_transform_time_and_frequency_domains_small.gif>

[5] <http://www.multiteide.es/wp-content/uploads/2016/10/Ejemplo2bis-768x677.jpg>

[6] <http://www.multiteide.es/wp-content/uploads/2016/10/Ejemplo4-768x767.jpg>