



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE ENGENHARIA ELÉTRICA E ELETRÔNICA
CURSO DE GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Luis Gustavo Hang Pereira

**Desenvolvimento de sistema de áudio espacial com a utilização de sinais monos com
funções de transferência relativas a cabeça**

Florianópolis

2024

Luis Gustavo Hang Pereira

Desenvolvimento de sistema de áudio espacial com a utilização de sinais monos com funções de transferência relativas à cabeça

Trabalho de Conclusão do Curso de Graduação em Engenharia Elétrica do Centro Tecnológico da Universidade Federal de Santa Catarina como requisito para a obtenção do título de Bacharel em Engenharia Elétrica.

Orientador: Prof. Sidnei Noceti Filho.

Coorientador: Prof. Walter Antônio Gontijo.

Florianópolis

2024

Ficha de identificação da obra

Pereira, Luis Gustavo Hang

Desenvolvimento de sistema de áudio espacial com a utilização de sinais monos com funções de transferência relativas à cabeça / Luis Gustavo Hang Pereira ; orientador, Sidnei Noceti Filho, coorientador, Walter Antônio Gontijo, 2024.

68 p.

Trabalho de Conclusão de Curso (graduação) - Universidade Federal de Santa Catarina, Centro Tecnológico, Graduação em Engenharia Elétrica, Florianópolis, 2024.

Inclui referências.

1. Engenharia Elétrica. 2. Audio tridimensional. 3. HRTF. 4. Processamento de áudio. I. Filho, Sidnei Noceti . II. Gontijo, Walter Antônio. III. Universidade Federal de Santa Catarina. Graduação em Engenharia Elétrica. IV. Título.

Luis Gustavo Hang Pereira

**Desenvolvimento de sistema de áudio espacial com a utilização de sinais monos
com funções de transferência relativa à cabeça**

Este Trabalho Conclusão de Curso foi julgado adequado para obtenção do Título de “Bacharel em Engenharia Elétrica” e aceito, em sua forma final, pelo Curso de Graduação em Engenharia Elétrica.

Florianópolis, 02 de dezembro de 2024.

Prof. Miguel Moreto, Dr.
Coordenador do Curso de Graduação em Engenharia Elétrica

Banca Examinadora:

Prof. Sidnei Noceti Filho, Dr.
Orientador
Universidade Federal de Santa Catarina

Prof. Walter Antônio Gontijo, Me.
Coorientador
Universidade Federal de Santa Catarina

Prof. Richard Demo Souza, Dr.
Universidade Federal de Santa Catarina

Prof. Walter Pereira Carpes Junior, Dr.
Universidade Federal de Santa Catarina

Prof. Stephan Paul, Dr
Universidade Federal de Santa Catarina

Este trabalho é dedicado aos meus aos meus queridos pais e aos meus amigos que fizeram parte durante esse período.

RESUMO

Este trabalho investiga o uso de Funções de Transferência Relacionadas à Cabeça (HRTFs) como método para geração de áudio tridimensional a partir de arquivos de áudios mono. O objetivo é posicionar virtualmente fontes sonoras em diferentes localizações espaciais, proporcionando uma experiência auditiva imersiva de forma prática e acessível. O método utiliza convolução das amostras de áudio de entrada com as HRTFs pré-carregadas e realiza cálculos de atenuação e atraso de sinais para simular o efeito direcional em cada orelha, caracterizado por parâmetros de Intensidade de Nível Diferencial (ILD) e Diferença de Tempo Interaural (ITD). Os testes envolveram sinais de frequência variados e ângulos de posições pré-definidas, aplicadas a áudio mono em formatos Wav e PCM, com análise qualitativa e quantitativa dos resultados. Observou-se que o método é eficaz em criar a sensação de movimentação de áudio, sendo perceptível o deslocamento de uma fonte sonora em função de diferentes posições angulares. Nos testes, ângulos de 90° e 270° apresentaram as maiores diferenças de ILD e ITD, indicando uma diferença clara de intensidade e tempo entre os canais esquerdo e direito conforme a posição espacial. Os resultados confirmam que a utilização de HRTFs facilita o processamento de áudio 3D sem necessidade de equipamentos de capturas estereofônicas complexas. Este trabalho contribui com um sistema eficiente e intuitivo para aplicação em áudios imersivos.

Palavras-chave: HRTF. Áudio 3D. Convolução, ILD, ITD, Processamento de áudio. Áudio imersivo.

ABSTRACT

This work investigates the use of Head-Related Transfer Functions (HRTFs) as a method for generating three-dimensional audio from mono audio files. The objective is to virtually position sound sources in different spatial locations, providing an immersive auditory experience in a practical and accessible way. The method employs convolution of input audio samples with pre-loaded HRTFs and performs signal attenuation and delay calculations to simulate directional effects for each ear, characterized by Interaural Level Difference (ILD) and Interaural Time Difference (ITD) parameters. Tests involved signals of varying frequencies and predefined position angles, applied to mono audio in Wav and PCM formats, with qualitative and quantitative analysis of the results. The method proved effective in creating a sense of audio movement, with the displacement of a sound source being perceptible according to different angular positions. In the tests, angles of 90° and 270° showed the greatest ILD and ITD differences, indicating a clear intensity and time difference between the left and right channels depending on the spatial position. The results confirm that using HRTFs facilitates 3D audio processing without requiring complex stereophonic recording equipment. This work contributes an efficient and intuitive system for applications in immersive audio.

Keywords: HRTF. 3D Audio. Convolution. ILD. ITD. Audio Processing. Immersive Audio.

LISTA DE FIGURAS

Figura 1 - Planos de referência.....	19
Figura 2 - Coordenadas polares.....	20
Figura 3 - Sistema auricular humano.....	22
Figura 4 - Representação da diferença de tempo interaural.....	24
Figura 5 - HRIR de 0° (a), de 45° (b) e 90° (c) na elevação de 0°, do KEMAR.....	27
Figura 6 - Magnitudes das HRTFs de 0° (a), 45° (b) e 90° (c) na elevação 0° do KEMAR.....	28
Figura 7 - HRIR de 0° (a), de 45° (b) e 90° (c) na elevação de 0° de um indivíduo do projeto Listen.....	29
Figura 8 - Magnitude das HRTFs de 0° (a), 45° (b) e 90° (c) na elevação 0° de um indivíduo do projeto Listen.....	29
Figura 9 - Setup para aquisições das HRIRs do projeto SADIE II.....	30
Figura 10 - Posições das medições das HRTFs do sujeito H4.....	31
Figura 11 - HRIR de 0° (a), de 45° (b) e 90° (c) na elevação de 0° do projeto SADIE II.....	32
Figura 12 - Magnitude das HRTFs de 0° (a), 45° (b) e 90° (c) na elevação 0° do projeto SADIE II.....	33
Figura 13 - Diagrama do projeto desenvolvido.....	37
Figura 14 - Bloco de diagramas da interface de usuário.....	40
Figura 15 - Interface do bloco “Upload da entrada”.....	41
Figura 16 - Interface da bloco “Quantidades de posições”.....	41
Figura 17 - Interface do bloco “Escolha dos ângulos para cada posição”.....	42
Figura 18 - Interface do bloco “Obtém as informações do PCM”.....	43
Figura 19 - Interface do bloco “Processamento de informações”.....	44
Figura 20 - Código em Python para a chamada da execução do processamento em C.....	45
Figura 21 - Interface do bloco “Realizar o download”.....	45
Figura 22 - Interface da “Saída áudio binaural”.....	46
Figura 23 - Exemplo de rotas da IDE.....	47
Figura 24 - Diagrama de blocos do processamento.....	48

Figura 25 - Vetor D_0 do ângulo 0°.....	50
Figura 26 - Vetor D_45 do ângulo 45°.....	50
Figura 27 - Vetor D_90 do ângulo 90°.....	51
Figura 28 - Todos os vetores utilizados no exemplo.....	51
Figura 29 - Espectrogramas original (a), do canal esquerdo (a), do canal direito (b) dos resultados do áudio da caminhada.....	56
Figura 30 - Comparação da entrada e saída de cada canal de um sinal senoidal de 100 Hz com a fonte na posição 90°.....	58
Figura 31 - Comparação da entrada e saída de cada canal de um sinal senoidal de 100 Hz com a fonte na posição 180°.....	59
Figura 32 - Comparação da entrada e saída de cada canal de um sinal senoidal de 100 Hz com a fonte na posição 270°.....	60

LISTA DE TABELAS

Tabela 1 – Medição das HRIRs do KEMAR.....	27
Tabela 2 – Resultado do sinal senoidal de 100 Hz.....	61
Tabela 3 – Resultado do sinal senoidal de 500 Hz.....	62
Tabela 4 – Resultado do sinal senoidal de 2000 Hz.....	62

LISTA DE ABREVIATURAS E SIGLAS

3D	Tridimensionalidade
CSS	Folhas de Estilo em Cascata
dB	Decibéis
dBFS	Decibéis em relação à escala total
FFT	Transformada rápida de Fourier
GCC	Coleção de Compiladores GNU
HRIR	Resposta ao impulso relacionado à cabeça
HRTF	Função de transferência relativa a cabeça
HTML	Linguagem de Marcação de Hipertexto
IDE	Ambiente de Desenvolvimento Integrado
ILD	Diferença de nível interaural
ITD	Diferença de tempo interaural
JND	Diferença mínima perceptível
MIT	Instituto de Tecnologia de Massachusetts
URL	Localizador Uniforme de Recursos

LISTA DE SÍMBOLOS

θ (Theta - representando direção azimutal)

ϕ (Phi - representando elevação zenital)

A (Amplitude do sinal)

s (Segundos)

t (Tempo)

Hz (Hertz)

F_s (Frequência de amostragem)

SUMÁRIO

1	INTRODUÇÃO.....	16
1.1	OBJETIVOS.....	17
1.1.1	Objetivo Geral.....	17
1.1.2	Objetivos Específicos.....	17
1.2	ESTRUTURA DO TRABALHO.....	17
2	FUNDAMENTAÇÃO TEÓRICA.....	18
2.1	CONCEITOS E DEFINIÇÕES.....	18
2.1.1	Áudio Monoaural e Binaural.....	18
2.1.2	Sistemas de Planos e Coordenadas.....	18
2.1.2.1	<i>Planos de referência.....</i>	<i>18</i>
2.1.2.2	<i>Coordenadas polares.....</i>	<i>19</i>
2.2	ANATOMIA DO SISTEMA AUDITIVO PERIFÉRICO DO SER HUMANO.....	21
2.2.1	Orelha externa, média e interna.....	21
2.2.2	Características binaurais ITD e ILD.....	22
2.3	LOCALIZAÇÃO DA FONTE SONORA.....	23
2.3.1	Cenário de escuta.....	23
2.3.2	Índice de tempo.....	23
2.3.3	HRTFs.....	24
2.3.4	HRIRs.....	25
2.4	BANCOS DE HRTFS.....	25
2.4.1	HRTFs Gardier e Martin.....	26
2.4.1.1	<i>Extração.....</i>	<i>26</i>
2.4.2	HRTFs Projeto Listen.....	28
2.4.2.1	<i>Extração.....</i>	<i>28</i>
2.4.3	HRTFs SADIE II.....	29
2.4.3.1	<i>Extração.....</i>	<i>30</i>
2.4.4	Conclusão dos banco de HRTFs.....	34

2.5	HTML E CSS.....	34
2.6	PYTHON.....	34
2.7	LINGUAGEM C.....	35
3	DESENVOLVIMENTO.....	36
3.1	VISÃO GERAL.....	36
3.2	ANÁLISE DAS HRTFS.....	37
3.2.1	Comparação das bases de dados.....	37
3.3	DESENVOLVIMENTO EM HTML E CSS.....	38
3.4	DESENVOLVIMENTO EM PYTHON.....	46
3.4.1	Aquisição das HRTFs.....	46
3.4.2	IDE.....	46
3.5	DESENVOLVIMENTO EM C.....	47
3.5.1	Criação dos vetores das posições.....	48
3.5.1.1	<i>Coleta de Informações.....</i>	49
3.5.1.2	<i>Cálculo de Parcelas de Tempo.....</i>	49
3.5.1.3	<i>Construção dos Vetores D.....</i>	49
3.5.2	Processamento.....	51
3.5.2.1	<i>Função main.....</i>	51
3.5.2.2	<i>Função openHRTFs.....</i>	52
3.5.2.3	<i>Função openFile.....</i>	52
3.5.2.4	<i>Função conv.....</i>	52
3.5.2.5	<i>Resumo do Fluxo de Processamento.....</i>	53
4	RESULTADOS.....	54
4.1	ANÁLISES.....	54
5	CONCLUSÃO.....	64
	REFERÊNCIAS.....	66

1 INTRODUÇÃO

O desenvolvimento da tecnologia voltado ao consumo de mídias tem se intensificado na criação de experiências imersivas, especialmente com o avanço de dispositivos utilizados em realidade virtual, jogos, entretenimento e simulação acústica. As indústrias musical e a de jogos têm sido fundamentais para o progresso da ciência e da tecnologia, com avanços significativos nas últimas décadas, em grande parte impulsionados por incentivos financeiros [1].

As mídias que oferecem experiências imersivas, como por exemplo videogames e plataformas de streamings para consumo de músicas, estão se consolidando no cenário do entretenimento. Em 2023, o setor games contou com mais de 3,3 bilhões de jogadores e faturou aproximadamente 188 bilhões de dólares [2]. No setor da música, o serviço de streaming mais popular conta com mais de 625 milhões de usuários, sendo que a maioria desses consome o conteúdo por meio de fones de ouvido [3].

O som é um dos sentidos que mais contribui para a sensação de imersão, especialmente por meio do áudio 3D ou áudio espacial, onde consegue-se ter a fonte sonora em uma certa posição — frequentemente obtida por meio de processamento digital de áudio. Neste contexto, o presente trabalho busca explorar técnicas de conversão de áudio mono para áudio espacial tridimensional (áudio 3D), visando proporcionar ao usuário uma experiência mais envolvente e realista, na qual seja possível identificar a localização da fonte sonora [4].

A técnicas de conversão de áudio mono para áudio tridimensional por meio de funções de transferência relativas à cabeça é comumente utilizado, como já apresentado (por exemplo, [5], [6] e [7]). O presente trabalho tem o objetivo de contribuir com a comparação de diferentes bancos de dados de funções de transferência relativas à cabeça, de modo a destacar o método de aquisição, além do desenvolvimento de um software que contribui na usabilidade das funções e permita a análise qualitativa e quantitativa dos resultados obtidos.

1.1 OBJETIVOS

Nas seções abaixo, estão descritos o objetivo geral e os objetivos específicos deste Trabalho de Conclusão de Curso.

1.1.1 Objetivo Geral

O objetivo principal deste Trabalho de Conclusão de Curso é realizar a conversão de sinais de áudio mono em áudio espacial tridimensional (áudio 3D).

1.1.2 Objetivos Específicos

- Extrair dados de bancos de dados utilizando a linguagem Python.
- Analisar bancos de dados de Funções de Transferência Relacionadas à Cabeça (HRTFs).
- Selecionar uma base de dados adequada para o desenvolvimento do projeto.
- Desenvolver uma interface do usuário em HTML e CSS.
- Implementar o processamento dos dados em Python.
- Realizar o processamento em linguagem C.
- Testar o processamento com diferentes fontes sonoras e combinações variadas de posições.
- Avaliar o desempenho e os resultados obtidos.

1.2 ESTRUTURA DO TRABALHO

Este trabalho está organizado da seguinte maneira:

No Capítulo 2, é apresentada a fundamentação teórica, oferecendo uma visão geral dos conceitos essenciais. O Capítulo 3 detalha o desenvolvimento da implementação. No Capítulo 4, são discutidos os resultados obtidos a partir da implementação. Por fim, o Capítulo 5 apresenta as considerações finais.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 CONCEITOS E DEFINIÇÕES

2.1.1 **Áudio Monoaural e Binaural**

O áudio monoaural é estruturado com apenas um canal, enquanto o áudio binaural é composto por dois canais. Esses dois canais podem ser idênticos, chamados de dióticos ou diferentes, sendo então denominados dicóicos. Em termos simplificados, todo sinal pode ser tratado como binaural, pois o ouvinte recebe o som em ambas as orelhas. No entanto, o termo estéreo binaural é normalmente reservado para sinais de dois canais que representam as características distintas de amplitude e fase em cada orelha, constituído, assim, um sinal dicóico [8].

2.1.2 **Sistemas de Planos e Coordenadas**

Para criar uma representação de áudio espacial, é essencial considerar sua tridimensionalidade. Existem dois modos principais para descrever essa tridimensionalidade: utilizando planos de referência e o sistema de coordenadas polares [9]. Sendo ambos complementares, o plano de referência indicando a posição da fonte sonora e o sistema de coordenadas indicando a direção em que se encontra a fonte sonora.

2.1.2.1 *Planos de referência*

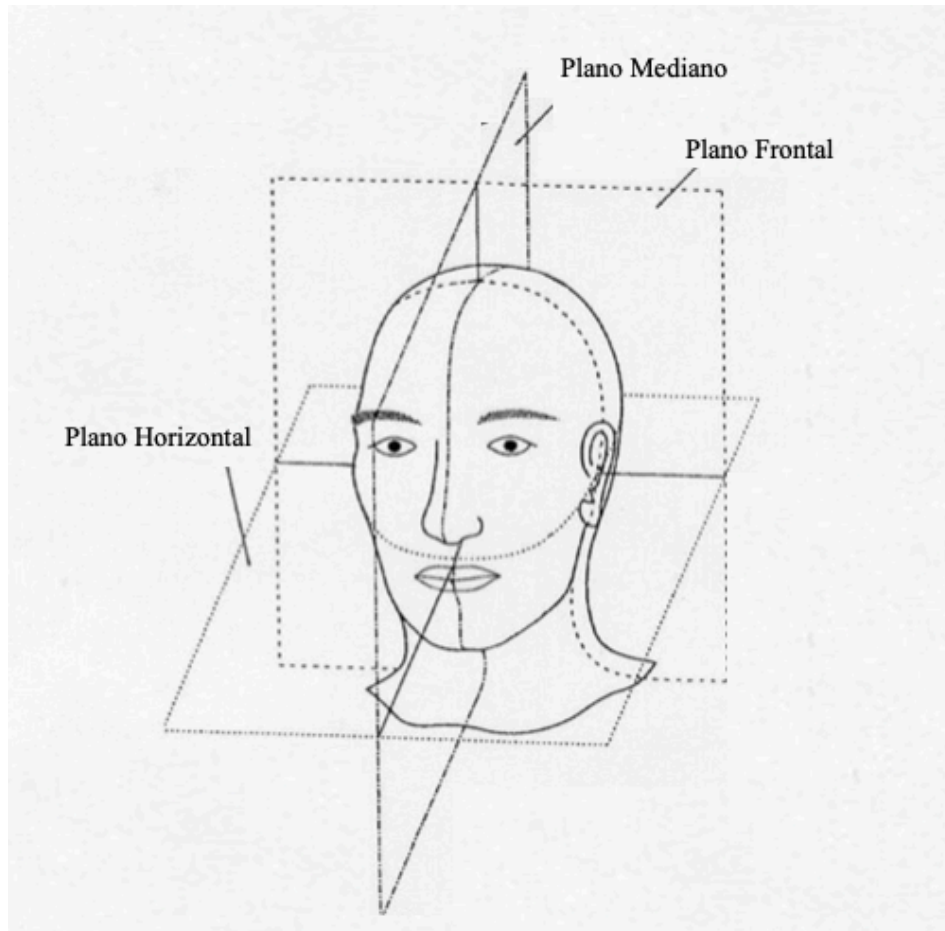
Como ilustrado na [Figura 1](#), os planos de referências são compostos por três planos principais: o plano frontal, o plano mediano e o plano horizontal.

O plano horizontal divide o espaço em hemisfério superior e inferior. Esse plano determina a direção da fonte sonora em relação ao ouvinte, indicando se o som vem da frente ou de atrás, e se está à esquerda ou à direita.

O plano frontal, também chamado de plano vertical, divide o espaço em hemisfério anterior e posterior. Esse plano define a elevação da fonte sonora, indicando se ele vem de cima, de baixo ou está na altura das orelhas.

Por fim, o plano mediano, ou plano sagital, divide o espaço em hemisfério esquerdo e direito. [10].

Figura 1 - Planos de referência



Fonte: [11].

2.1.2.2 Coordenadas polares

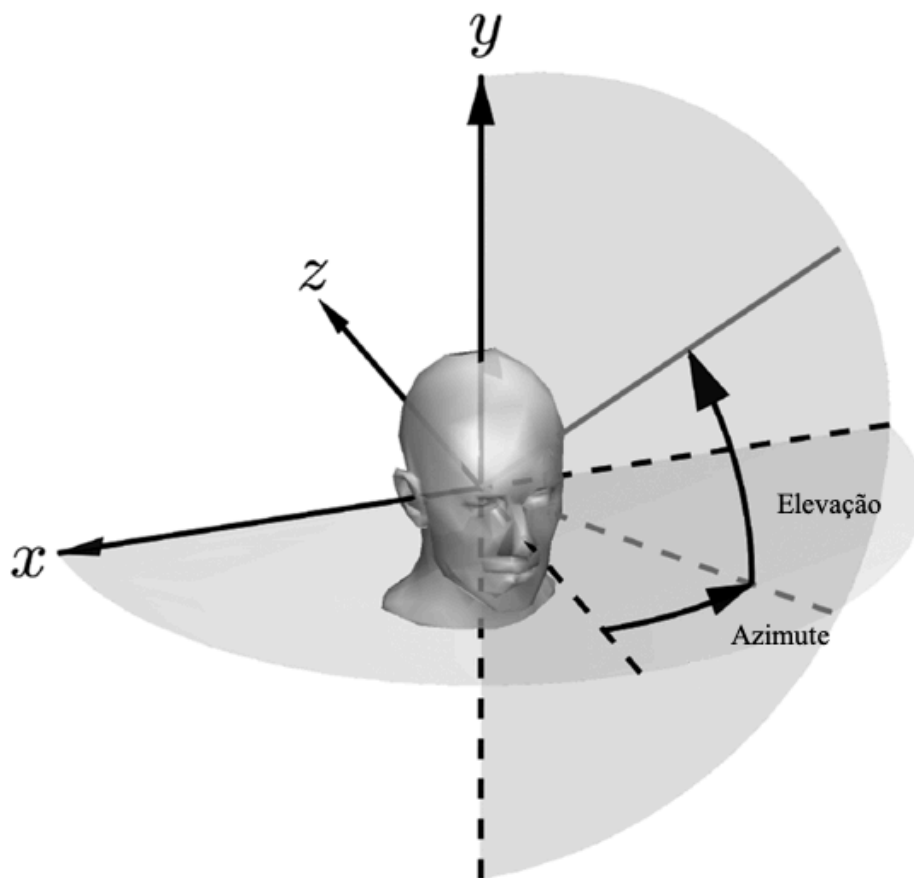
O sistema de coordenadas polares, ilustrado na [Figura 2](#), é comumente utilizado para definir a posição da fonte sonora em um áudio espacial, descrevendo a direção de qualquer ponto por meio de dois ângulos: azimute e elevação.

O ângulo azimutal (θ) refere-se ao plano horizontal, definindo a direção da fonte sonora em relação ao receptor ou em relação a algum ponto, e.g. a origem do sistema de coordenadas. Esse ângulo varia de 0° a 359° , com 0° representando a

direção frontal, 180° a direção posterior, 90° a direção da esquerda e 270° a direção direita.

O ângulo da elevação (ϕ) indica a altura da fonte sonora. A elevação varia de -90° a 90° , onde -90° representa a direção inferior (na direção dos pés, se o ouvinte estiver em pé), 90° representa a direção superior (acima da cabeça) e 0° a direção ao nível das orelhas [10].

Figura 2 - Coordenadas polares



Fonte: [12].

2.2 ANATOMIA DO SISTEMA AUDITIVO PERIFÉRICO DO SER HUMANO

Para realizar o processamento de áudio que proporciona uma sensação espacial, é fundamental compreender de uma forma básica o funcionamento da audição humana e seus aspectos fisiológicos.

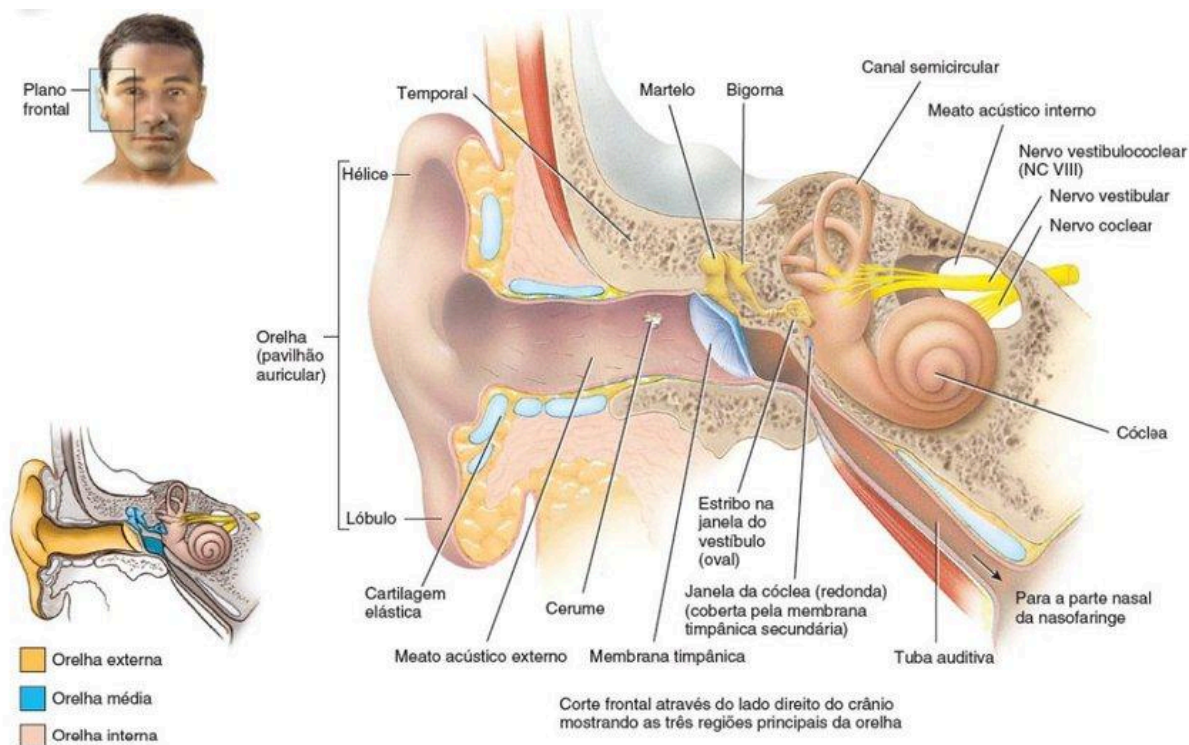
2.2.1 Orelha externa, média e interna

A audição humana envolve a captação de mudanças de pressão sonora, que são percebidas pelas orelhas, composta de três partes principais: a externa, a média e a interna. A parte externa possui uma parte visível chamada de pavilhão auditivo, ou pinna ([Figura 3](#)) e um canal auditivo que leva até o tímpano ([Figura 3](#)). Essa região funciona como um filtro direcional para o interior da orelha.

A orelha média é responsável pela transmissão do som. Ele contém os ossículos — martelo, bigorna e estribo, como ilustrado na [Figura 3](#) —, que amplificam certas faixas de frequências e conduzem as vibrações sonoras para a orelha interna.

Por fim, a orelha interna contém a cóclea como pode ser observado na [Figura 3](#), que transforma variações de pressão sonora no fluido coclear em impulsos elétricos, codificando pressão sonora RMS, frequência e para baixas frequência a fase do sinal [13].

Figura 3 - Sistema auricular humano



Fonte: [14].

2.2.2 Características binaurais ITD e ILD

O nervo auditivo, inicia na cóclea e termina nos córtices auditivos ([Figura 3](#)), onde ocorre a transformação do sinal. A integração das informações de ambas as orelhas permite a extração de características binaurais, como a diferença de tempo interaural (ITD) e a diferença de nível interaural (ILD).

A diferença de tempo interaural (ITD) é processada por células conhecidas como detectores de coincidências. Essas células funcionam como comparador, analisando os sinais de cada orelha, buscando correlação e determinando a diferença temporal entre eles.

Já a diferença de nível interaural (ILD) é processada de forma semelhante ao ITD, mas envolve um aumento do nível inibitório. Esse nível inibitório é a atividade neural interaural necessária para inibir complementarmente a resposta entre os neurônios. Assim, o ILD representa o esforço necessário para atingir essa inibição completa da resposta neural [13].

2.3 LOCALIZAÇÃO DA FONTE SONORA

2.3.1 Cenário de escuta

O cenário de escuta refere-se ao ambiente em que a fonte sonora e o ouvinte estão inseridos. Neste trabalho, será considerado um cenário de escuta padrão, com uma única fonte sonora em um campo livre, onde não há outros objetos ou superfícies que possam refletir o som. Câmaras anecoicas são comumente utilizadas em ensaios laboratoriais para simular esse tipo de ambiente e facilitar a aquisição de dados sobre a localização de fontes sonoras [13].

2.3.2 Índice de tempo

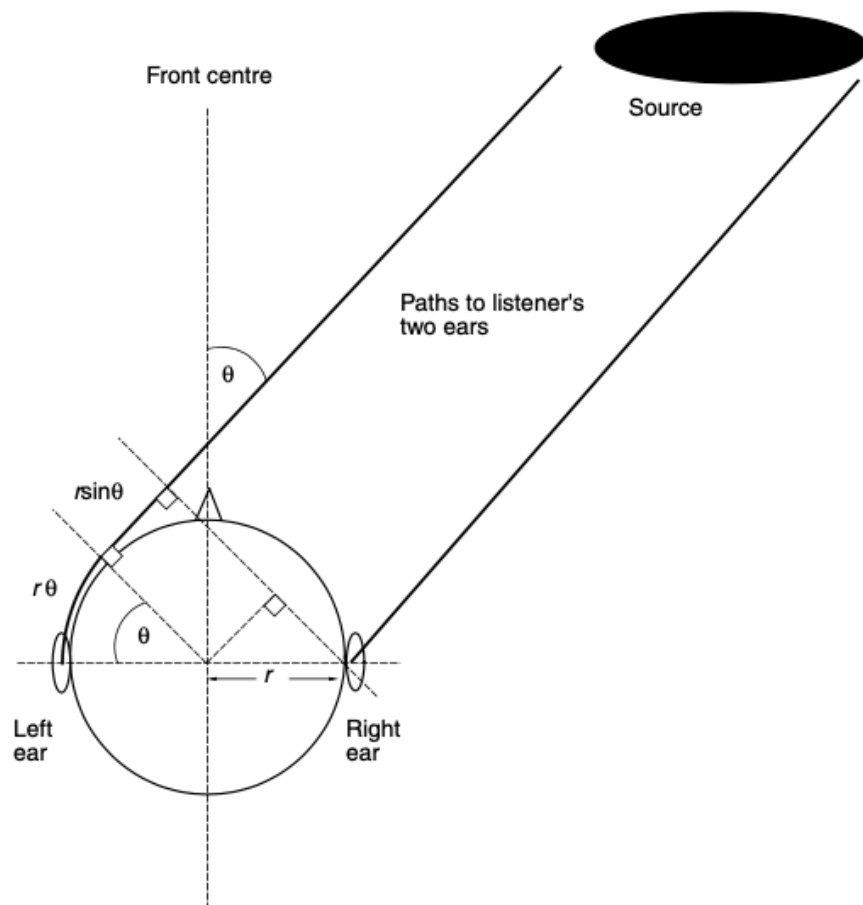
Quando uma fonte sonora não está posicionada a 0° no plano horizontal, ocorre uma diferença temporal da chegada do som a cada orelha, o que resulta na ITD. Esse tempo de diferença depende do ângulo da fonte sonora e da distância entre as orelhas, relacionado ao raio da cabeça do ouvinte, como ilustrado na [Figura 4](#). A fórmula de ITD (Equação 1) para fontes suficientemente distantes da cabeça, assumindo frentes de ondas planas, pode ser escrita da seguinte maneira:

$$ITD = \frac{r(\theta + \sin\theta)}{c} \quad (1)$$

Onde r é o raio da cabeça em metros, θ é o ângulo da posição no plano horizontal, e c é a velocidade do som (aproximadamente 340 m/s em um cenário de escuta padrão). A partir da Equação 1, observa-se que quando o ângulo é de 0° , o ITD resultará em zero. Já com a fonte sonora a 90° , o ITD atinge seu valor máximo; se o raio da cabeça for de 0,08 m, o ITD é aproximadamente de 0,605 milissegundos. Essa diferença temporal permite que o sistema auditivo consiga identificar a direção da fonte sonoras de baixas frequências, pois o sistema auditivo consegue distinguir a diferença temporal para frequências abaixo de 1000 Hz aproximadamente. No entanto, apenas com o ITD, não é possível distinguir se o som vem de 0° ou 180° , nem a elevação da fonte. É necessário analisar outras características, como ILD, padrões

espectrais (que são as modificações na amplitude e na frequência do som devido à interação com o pavilhão auricular, orelhas, cabeça e ombros), movimentação da cabeça e também reflexões ambientais [8].

Figura 4 - Representação da diferença de tempo interaural



Fonte: [8].

2.3.3 HRTFs

O sistema auditivo atua como um filtro para os sinais provenientes de diferentes posições da fonte sonora, resultando em uma relação de filtragem distinta para cada posição e orelha. O processo é descrito, formalmente, por HRTFs (do inglês *head-related transfer functions*) [13].

As HRTFs dependem da anatomia individual, pois os pavilhões auriculares são como as impressões digitais, cada indivíduo possui uma combinação única. Portanto, cada indivíduo filtra o som de maneira particular, gerando suas próprias combinações

de HRTFs. No entanto, estudos mostram que certas características das HRTFs são comuns e possuem uma importância maior para a localização da fonte sonora, o que permite a criação de modelos que abrangem uma maior quantidade de pessoas.

Entre as características importantes para as HRTFs está a ressonância da concha, que ocorre na cavidade da concha, sendo considerada responsável pela percepção de que o áudio está localizado fora da cabeça [12]. Outro fator é o tamanho da cabeça, que atua como uma barreira significativa para as componentes de alta frequência do som. Além disso, outras partes do corpo, como o ombro e o tronco, geram reflexões que alteram o espectro do som. Todos esses fatores resultam em HRTFs únicas para cada posição e ângulo de incidência das ondas sonoras, além de ser individualizada para cada pessoa [8].

2.3.4 HRIRs

As respostas ao Impulso Relacionadas à Cabeça (HRIRs, do inglês *head-related impulse response*) são respostas de impulsos associadas às HRTFs para uma posição espacial específica. Em outras palavras, as HRIRs representam a resposta de um impulso do caminho de propagação entre fonte e membranas timpânicas, ou um ponto de referência no canal auditivo [15].

2.4 BANCOS DE HRTFS

Neste trabalho, serão comparados três diferentes bancos de HRTFs: Gardner e Martin, Listen e SADIE II. Cada um desses bancos oferece conjuntos distintos de HRIRs, fornecendo uma base diversificada para a escolha da opção mais adequada ao projeto, considerando aspectos como resolução espacial, faixa de frequência e metodologia de captura.

A escolha desses bancos se dá por serem bancos que estão disponíveis de forma gratuitas e suas características, o banco de HRTFs de Gardier e Martin foi escolhido devido sua importância para o desenvolvimento na área, o projeto Linten pela diversidade de usar um grupo grande para a aquisição e o SADIE II devido ser um projeto recente e por se destacar no número de medições.

2.4.1 HRTFs Gardier e Martin

A aquisição das HRIRs das HRTFs do conjunto de Gardier e Martin [16] foi realizada no MIT *Media Lab Perceptual Computing*, utilizando um manequim da KEMAR (*Knowles Electronic Manikin for Acoustic Research*).

As medições foram feitas a partir de respostas ao impulso registradas nas orelhas direita e esquerda do manequim, com um alto-falante *Realistic Optimus Pro* posicionado a uma distância de 1,4 metros.

As HRIRs foram amostradas de 0° a 180° no plano horizontal e de -40° a 90° de elevação, totalizando 710 posições de captura [16].

2.4.1.1 Extração

O banco de dados das HRTFs foi obtido por meio do portal do laboratório (site do MIT). Após a extração, foram realizados testes utilizando Python. O banco possui uma estrutura organizada em pastas, com elevações variando de -40° a 90° em incremento de 10° . Em cada pasta, encontra-se o conjunto de HRIRs para diferentes posições. Na [Figura 5](#), são exibidos os HRIRs das posições 0° (a), 45° (b) e 90° (c), todas com elevação de 0° .

No banco de dados é disponibilizada as HRIRs até 180° (exceto na elevação de 50° , onde foi realizado até 176°); ao realizar a transformada rápida de Fourier (FFT) e obter as HRTFs. É possível estender as posições de 180° a 360° invertendo o conjunto de HRIR das orelhas direita e esquerda, pois o banco de dados tem a característica de ser simétrico. Na Tabela 1 se encontra o passo utilizado para a realização das medições para cada elevação, além do número das medições.

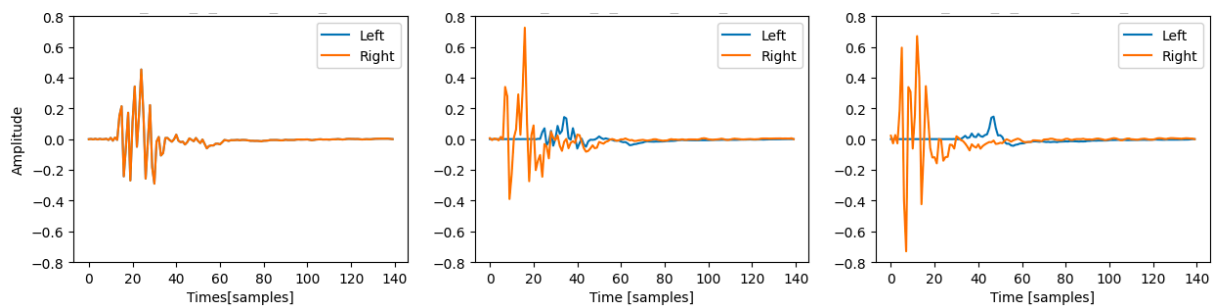
Tabela 1 – Medição das HRIRs do KEMAR.

Elevações	Número de HRIRs	Passo
-40°	29	6° a 7°
-30°	31	6°
-20°	37	5°
-10°	37	5°
0°	37	5°
10°	37	5°
20°	37	5°
30°	31	6°
40°	29	6° a 7°
50°	23	8°
60°	19	10°
70°	13	15°
80°	7	30°
90°	1	-

Fonte: Autor.

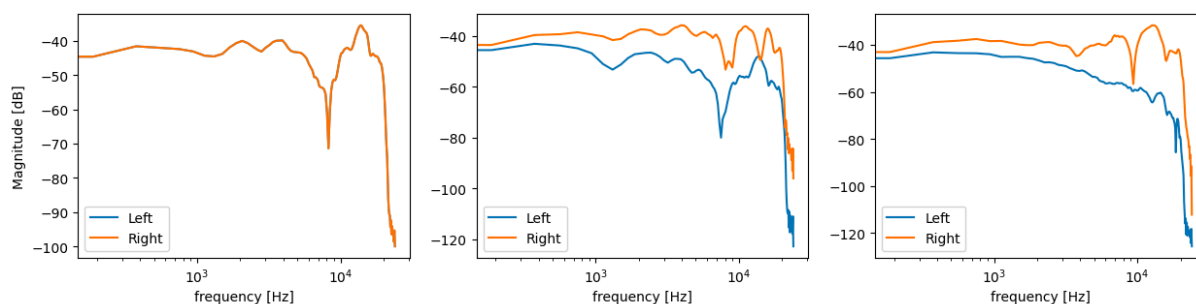
Após a extração dos HRIRs, aplica-se a FFT para obter as HRTFs, ilustradas na [Figura 6](#). As HRIRs permitem realizar convoluções com áudios mono, resultando em um áudio posicionado conforme a posição correspondente da HRIR [16].

Figura 5 - HRIR de 0° (a), de 45° (b) e 90° (c) na elevação de 0°, do KEMAR.



Fonte: Autor.

Figura 6 - Magnitudes das HRTFs de 0° (a), 45° (b) e 90° (c) na elevação 0° do KEMAR.



Fonte: Autor.

2.4.2 HRTFs Projeto Listen

As HRTFs do projeto Listen são disponibilizadas pela IRCAM (*Institute for Research and Coordination in Acoustics/Music*) [17], em colaboração com o programa IST (*Information Society Technologies*) da *European Commission's Fifth Framework*, que visa integrar e fortalecer o espaço de pesquisa europeu [18]. Essa parceria conta também com a AKG, empresa no setor da indústria de áudio, que contribui para pesquisas na área de áudio [19].

As HRIRs foram obtidas em uma sala anecoica (um ambiente sem reverberação) utilizando microfones posicionados nos canais auditivos de pessoas sentadas em uma cadeira giratória. O projeto disponibiliza 51 conjuntos de HRIRs, cada um correspondente a um participante. Os ângulos de posição variam de -45° a 90° , com incremento de 15° . O número de posições horizontais varia de 24 (para elevações de -45° a 45°) até apenas 1 (na elevação de 90°) [17].

2.4.2.1 Extração

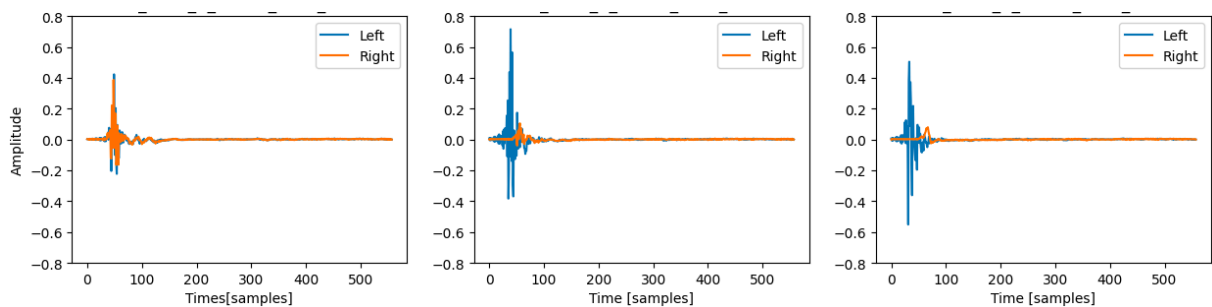
Cada conjunto contém 187 posições. Na [Figura 7](#), são apresentadas as HRIRs das posições 0° (a), 45° (b) e 90°, todas na elevação de 0°.

Especificamente nas elevações de -45° a 45° , a amostragem foi feita com passos de 15° , abrangendo posições de 0° a 345°, totalizando 24 HRIRs. Na posição de 60°, a amostragem ocorreu em passos de 30° , totalizando 12 HRIRs de 0° a 330°. Na posição de 75°, as medições foram feitas em passos de 60° , totalizando 6 HRIRs de 0°

a 300°. Por fim, na elevação de 90°, foi realizada apenas uma medição na posição 0°, resultando em um total de 187 pontos [17].

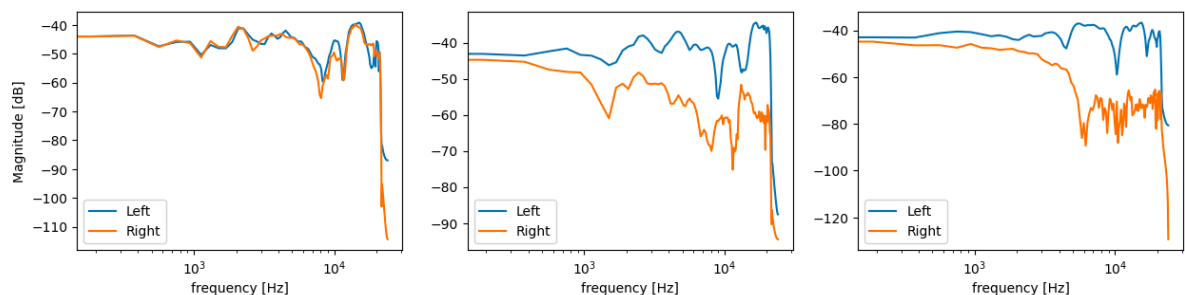
Após a obtenção das HRIRs, realiza-se a FFT para extrair as HRTFs, conforme ilustrado na [Figura 8](#).

Figura 7 - HRIR de 0° (a), de 45° (b) e 90° (c) na elevação de 0° de um indivíduo do projeto Listen.



Fonte: Autor.

Figura 8 - Magnitude das HRTFs de 0° (a), 45° (b) e 90° (c) na elevação 0° de um indivíduo do projeto Listen.



Fonte: Autor.

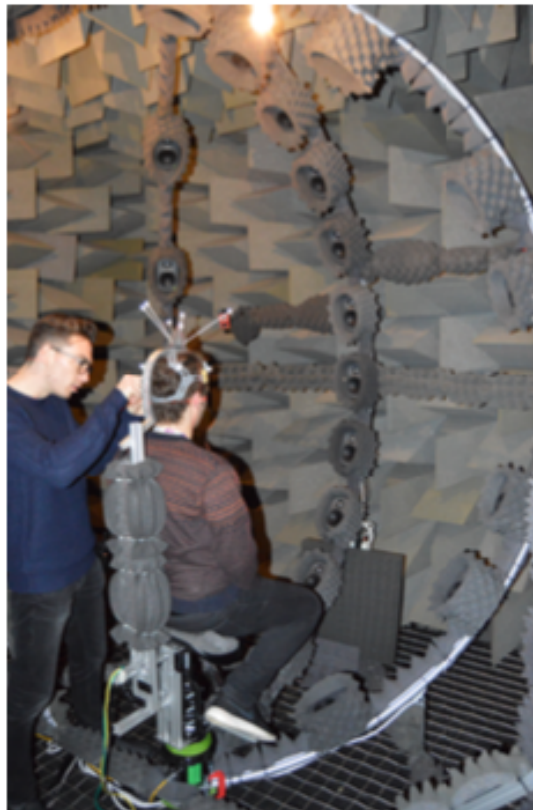
2.4.3 HRTFs SADIE II

O banco de dados das HRTFs do SADIE II [20] foi obtido através de medições de HRIRs realizadas em uma sala anecoica no laboratório de áudio da Universidade de York, no Reino Unido. Para a aquisição, foram medidas as HRIRs de vinte sujeitos, sendo 15 homens, 1 mulher, 2 não-binários e 2 manequins (um KU100 e outro KEMAR). A configuração para a extração das HRIRs consiste em três arcos semicirculares verticais estáticos, cada um separado por 45° na posição horizontal. Os

participantes eram posicionados no centro dos arcos, como ilustrado na [Figura 9](#), em um banco motorizado que girava em torno de seu próprio eixo. Nos arcos, havia 23 caixas de som dispostas a um raio de 1,2 metros.

As elevações variam de -75° a 90° , com um passo de 15° , incluindo uma medição adicional a -81° para permitir a obtenção computacional do ângulo de -90° , através de interpolação das medições realizadas. Para a aquisição das posições azimutais, foram realizados giros nos voluntários, fazendo paradas para permitir a coleta dos dados. Nos experimentos com manequins, a resolução foi de 1° , resultando em 399 paradas e gerando 8802 medições, levando cerca de 3 horas. Para os grupos de voluntários, uma configuração utilizou uma resolução de 5° , com 127 paradas e 2818 medições, levando aproximadamente 1 hora 15 minutos. Um segundo grupo operou com uma resolução 10° , totalizando 95 paradas e gerando 2114 medições em um tempo aproximadamente de 1 hora [21].

Figura 9 - Setup para aquisições das HRIRs do projeto SADIE II.



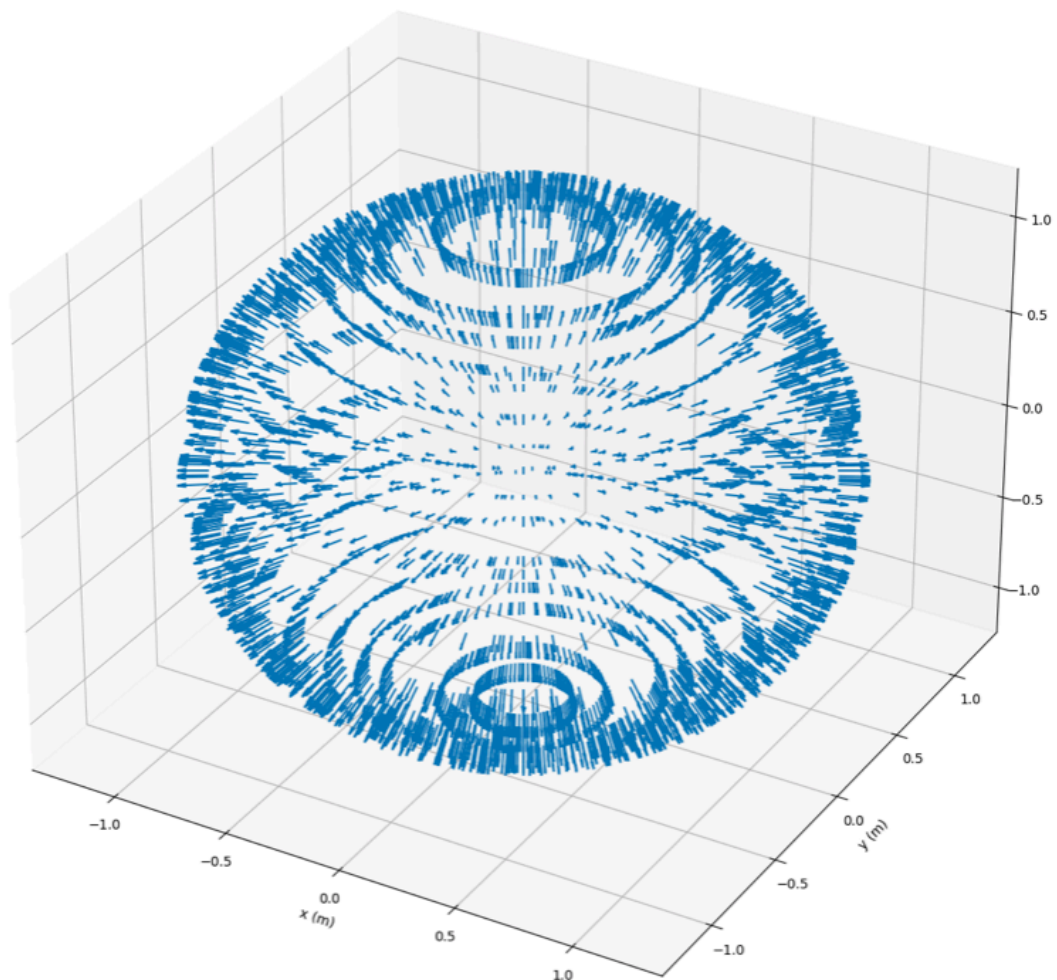
Fonte: [21].

2.4.3.1 Extração

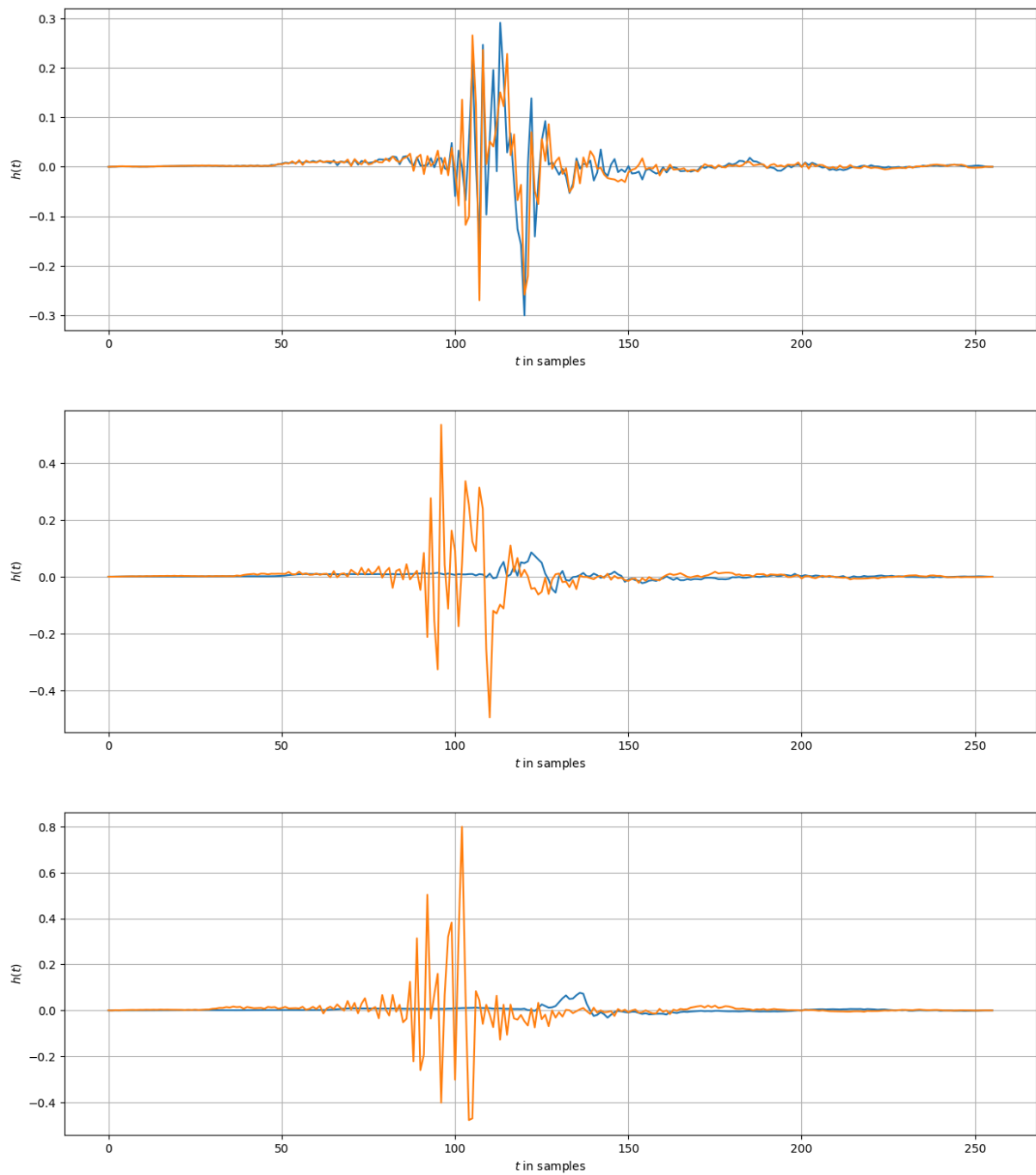
As bases de dados da SADIE II são compostas por 20 conjuntos de medições obtidas de diferentes sujeitos, sendo dois manequins e 18 humanos. Para a realização da extração, foi escolhido o sujeito H4, que é um humano que foi coletado um total de 2818 posições [21]. As posições podem ser visualizadas na [Figura 10](#).

Na [Figura 11](#), são apresentadas as HRIRs das posições de 0° (a), 45° (b) e 90° , todas na elevação de 0° . Após a obtenção das HRIRs, é realizada a FFT para extrair as HRTFs, que são mostradas na [Figura 12](#).

Figura 10 - Posições das medições das HRTFs do sujeito H4.

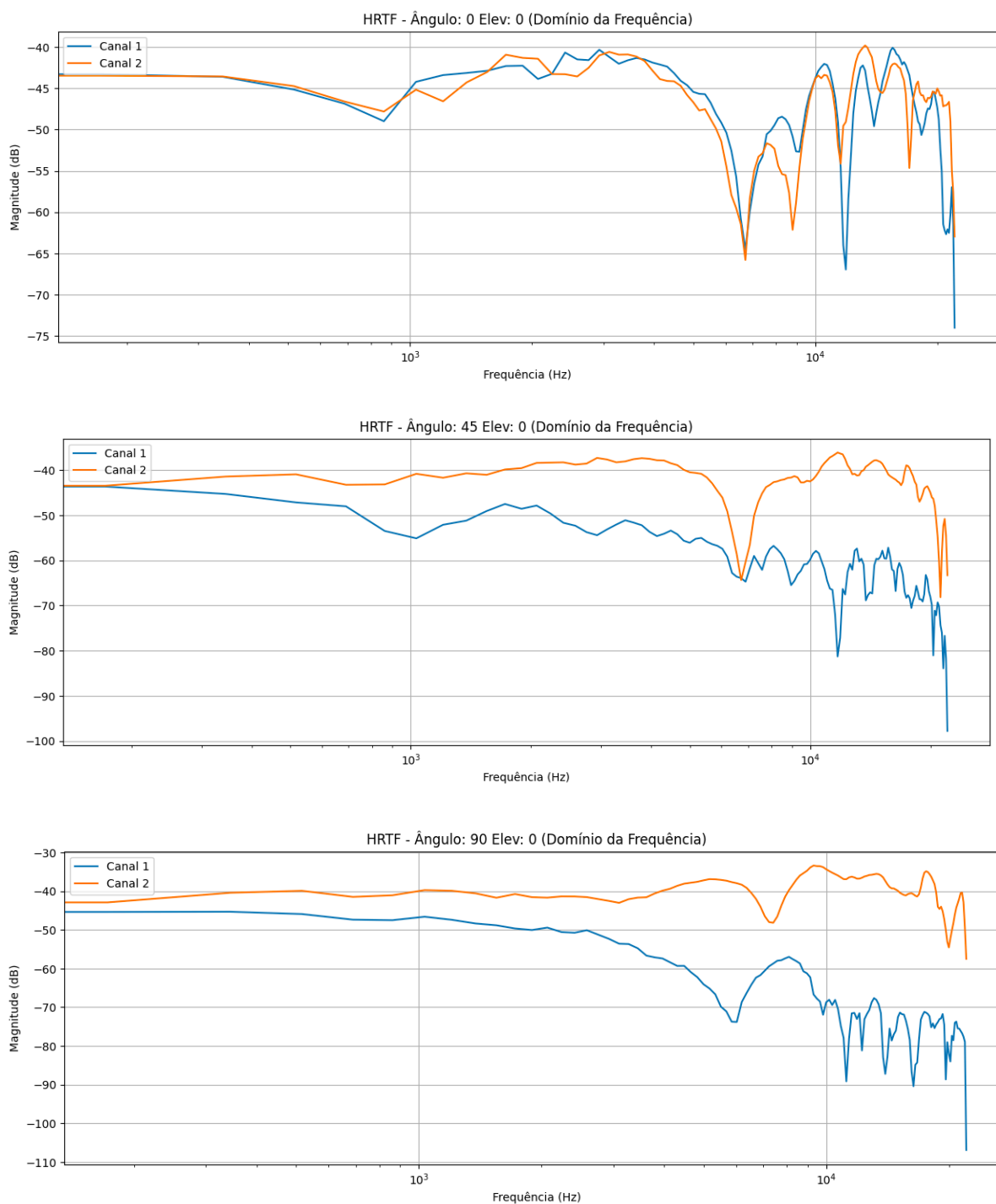


Fonte: Autor.

Figura 11 - HRIR de 0° (a), de 45° (b) e 90° (c) na elevação de 0° do projeto SADIE II.

Fonte: Autor.

Figura 12 - Magnitude das HRTFs de 0° (a), 45° (b) e 90° (c) na elevação 0° do projeto SADIE II.



Fonte: Autor.

2.4.4 Conclusão dos banco de HRTFs

No desenvolvimento deste projeto, será realizada uma comparação detalhada dos principais aspectos das bases de dados de HRTFs apresentados anteriormente. O objetivo é identificar a base que melhor se adapta às necessidades específicas do projeto, levando em consideração fatores como a exatidão das medições, a diversidade das posições e a aplicabilidade dos dados na implementação prática.

2.5 HTML E CSS

HTML e CSS desempenham um papel fundamental na construção da interface de usuário e na apresentação visual de páginas *web*.

HTML (*HyperText Markup Language*) é a linguagem padrão utilizada para criar a estrutura de páginas *web*, definindo os elementos básicos que compõem o conteúdo, como títulos, parágrafos, *links*, imagens e outros componentes interativos. Cada elemento HTML é organizado em uma hierarquia de *tags* que estruturam o documento, permitindo que o navegador compreenda e exiba corretamente o conteúdo [22].

CSS (*Cascading Style Sheets*) é a linguagem empregada para definir a aparência e o *layout* de uma página *web*. Enquanto o HTML cria a estrutura, o CSS estiliza essa estrutura, controlando cores, fontes, espaçamentos, tamanhos e disposições dos elementos na tela [22].

Com o uso de HTML e CSS, será realizada a estruturação visual do projeto, possibilitando ao usuário inserir um áudio e realizar as configurações para obter o efeito de áudio 3D.

2.6 PYTHON

Python é a segunda linguagem de programação mais utilizada, conhecida por sua versatilidade e potência, especialmente nas áreas de ciências de dados e processamento de sinais. Amplamente empregada para análise de informações em diversos setores [23]. No contexto das HRTFs, Python oferece ferramentas robustas para o processamento e análise de sinais de áudio.

Python, é amplamente utilizado pois a manipulação de *arrays* e dados [24] se tornando fácil com bibliotecas como NumPy, que permitem a leitura, limpeza e organização dos bancos de dados [25]. Além disso, bibliotecas especializadas, como librosa e SciPy fornecem funcionalidades para a análise de sinais de áudio, incluindo filtragem, transformações espectrais e extração de características, essenciais para compreender a variação da resposta auditiva em diferentes condições [26].

A integração dessas capacidades analíticas com plataformas *web*, por meio de *frameworks* como Flask [27], possibilita que as análises de HRTFs sejam realizadas diretamente em uma interface online, acessível a pesquisadores e desenvolvedores de tecnologias de áudio. Isso cria um ambiente interativo dentro de uma IDE desenvolvida para o site. Esse tipo de aplicação será relevante no projeto, pois visa personalizar experiências auditivas com um sistema de áudio 3D e realidade aumentada, permitindo a adaptação das HRTFs conforme as características individuais do usuário.

2.7 LINGUAGEM C

A linguagem C é uma das linguagens de programação mais antigas e influentes, desenvolvida inicialmente na década de 1970 por Dennis Ritchie. É uma linguagem de propósito geral e de baixo nível, que oferece controle direto sobre o *hardware*, além de ser extremamente eficiente em termos de desempenho de memória.

Reconhecida por seu desempenho otimizado e pelo controle preciso que proporciona sobre o *hardware*, a linguagem C torna-se uma escolha ideal para o processamento de dados do projeto e aplicações que exigem alta eficiência. Ao lidar com funções matemáticas complexas, que descrevem como o som é modificado pelas características físicas, a linguagem C permite uma manipulação direta da memória. Isso otimiza o uso de recursos e garante uma execução mais rápida em comparação com linguagens de nível mais alto, como o Python [28].

3 DESENVOLVIMENTO

Neste capítulo, será apresentada inicialmente uma visão geral do projeto, seguida da escolha das HRTFs mais adequadas. Em sequência, será abordado o desenvolvimento em Python, explorando suas capacidades para processamento de sinais e análise de dados. Também será discutido o uso de HTML e CSS para a criação da interface de usuário, proporcionando uma estrutura visual para o projeto. Por fim, será descrito o desenvolvimento na linguagem C, destacando seu papel essencial para o processamento eficiente de dados.

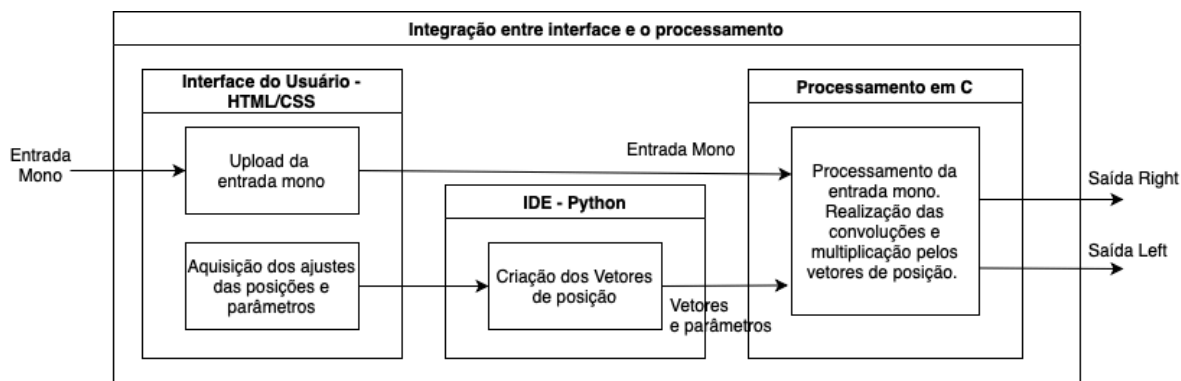
3.1 VISÃO GERAL

O objetivo deste projeto é converter sinais de áudio mono em um áudio espacial 3D, proporcionando uma experiência imersiva. Para o desenvolvimento, utilizam-se as seguintes ferramentas de software: Python, HTML, CSS e linguagem C. As tecnologias HTML e CSS são aplicadas na criação da interface com o usuário, enquanto o Python é utilizado para construir o ambiente de desenvolvimento integrado. Já a linguagem em C desempenha um papel fundamental no processamento do sinal, como ilustra a [Figura 13](#).

Na interface do usuário, o *front-end* foi desenvolvido com HTML e CSS, onde o usuário poderá inserir o áudio mono como entrada, definir parâmetros específicos e ajustar as configurações de posição para definir a trajetória da fonte sonora. O *back-end*, construído em Python com o *framework* Flask, é responsável por preparar os vetores e demais parâmetros, que serão então enviados ao processamento em C juntamente com a entrada mono.

A etapa central do processamento consiste na convolução entre o sinal de entrada mono e diferentes HRIRs, permitindo a conversão do áudio mono em áudio 3D espacializado.

Figura 13 - Diagrama do projeto desenvolvido.



Fonte: Autor.

3.2 ANÁLISE DAS HRTFS

O desenvolvimento do projeto teve início com a seleção das HRTFs que apresentaram os melhores resultados para diferentes ângulos de localização da fonte sonora. Para isso, foi realizada uma análise comparativa entre os bancos de dados de HRTFs discutidos na seção 2.4. Em Python, foi aplicada as HRIRs de cada banco de HRTFs em processos de espacialização sonora similares, usando os mesmos áudios com as mesmas seleções de direção, o que permitiu avaliar o desempenho de cada conjunto sob condições controladas.

Essa abordagem possibilitou uma comparação direta dos resultados obtidos, com base na precisão com que cada HRTFs reproduzia a percepção espacial do som, se diferentes pessoas obtiveram a sensação da direção de forma correta, conseguindo identificar de forma clara a posição da fonte sonora. Como critério de escolha, priorizou-se a capacidade de cada HRTF em garantir uma representação espacial precisa e consistente, culminando na seleção do banco de dados dados que melhor atendeu os objetivos do projeto.

3.2.1 Comparação das bases de dados

Para a escolha da base de HRTFs a ser utilizada no projeto, foram realizadas análises comparativas considerando aspectos como o método de aquisição das HRIRs, o número de posições mapeadas disponíveis e a capacidade qualitativa de distinguir

diferentes posições das fontes sonoras. Foram selecionadas as três bases de dados de HRTFs para os testes: Gardier e Martin, Listen e SADIE II, cada uma com características distintas quanto ao modo de aquisição, quantidade de amostras e tratamentos.

O primeiro critério de análise foi o método de aquisição. No projeto de Gardier e Martin, as HRIRs foram mensuradas para um receptor tipo manequins, tornando as respostas auditivas mais genéricas, sendo mais adequadas para uma média populacional, no entanto menos detalhadas [16]. O projeto Listen realizou a coleta das HRIRs com 51 sujeitos humanos, trazendo uma diversidade que enriquece o detalhamento das HRTFs, embora cada sujeito apresente características fenotípicas distintas, resultando em variações individuais nas HRTFs [17]. Por fim, a SADIE II se destaca ao incluir tanto manequins quanto 18 sujeitos humanos, proporcionando uma diversidade significativa nas amostragens das HRTFs [20].

Quanto à resolução espacial, o projeto SADIE II apresenta uma vantagem notável, com 8802 posições para os manequins, para os participantes humanos, dois grupos: um com 2818 e outro com 2114 posições [21]. O projeto Gardier e Martin contém 710 posições [16], enquanto o projeto Listen conta com 187 posições [17].

E para análise qualitativa, foram colocados diversos áudios convoluídos com as HRIRs e foram analisados por diferentes indivíduos quais bancos de dados proporcionaram melhores resultados, e um dos principais pontos seria distinguir a posição 0° com a posição 180°, para superar a chamada *front-back confusion* (em uma tradução livre, confusão entre a posição frontal e posterior). E percebeu que os projetos que utilizam humanos para aquisição das HRIRs obtiveram um resultado melhor, portanto a Listen e SADIE II se destacam nesse quesito.

Assim, o projeto SADIE II apresentou o melhor resultado, destacando-se pela diversidade de participantes para aquisição, grande número de posições e também o bom desempenho no processamento de áudio espacializado.

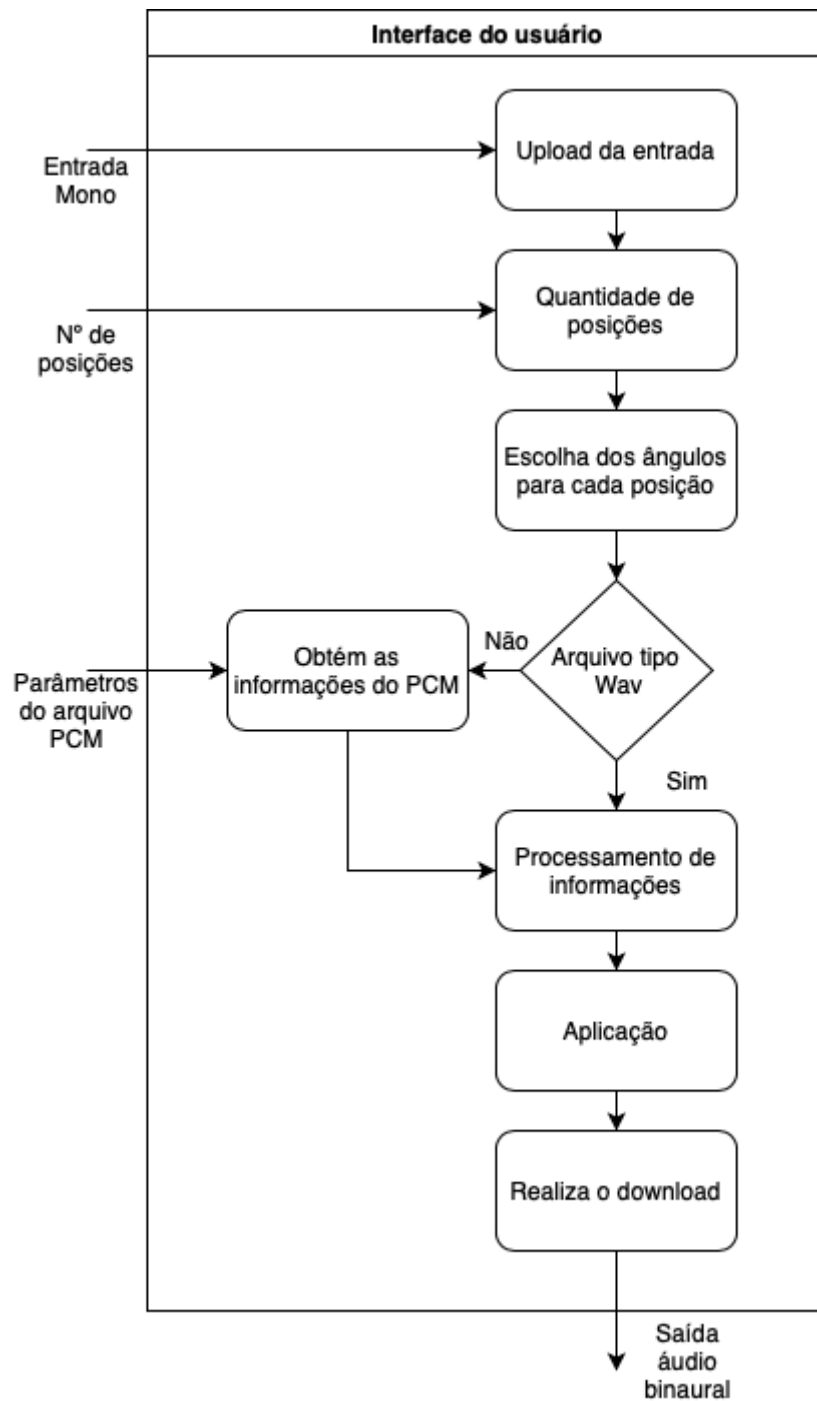
3.3 DESENVOLVIMENTO EM HTML E CSS

Os templates HTML, junto com o CSS, desempenham um papel essencial na aplicação desenvolvida em Flask. Esses arquivos são renderizados dinamicamente com dados específicos antes de serem enviados ao usuário, o que permite a criação de páginas *web* dinâmicas e com conteúdo variável. Essa abordagem torna possível a

personalização da interface com base nas interações do usuário e nas informações fornecidas.

Na sequência, é apresentada uma breve descrição das diferentes interfaces tratadas na IDE em Python. Na [Figura 14](#), é mostrado o diagrama de blocos da interface do usuário, que ilustra as etapas pelas quais o usuário passa para inserir os dados e definir os parâmetros necessários para o processamento.

Figura 14 - Bloco de diagramas da interface de usuário.

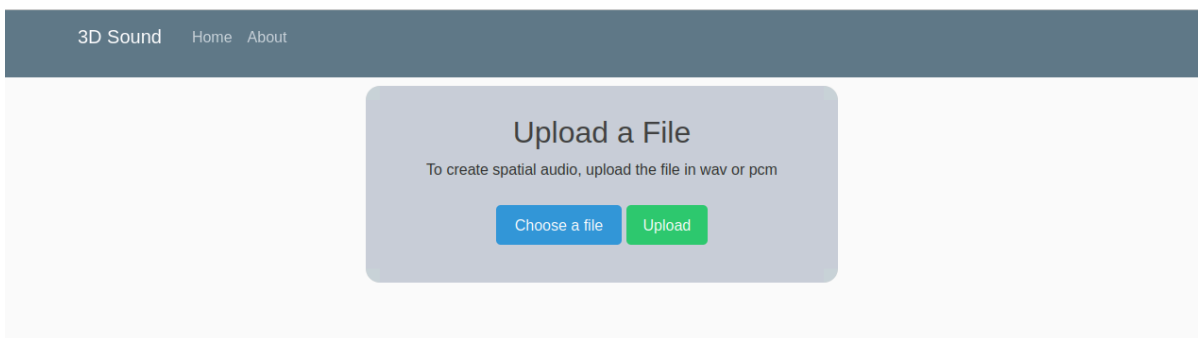


Fonte: Autor.

No bloco de “Upload da Entrada” permite ao usuário enviar um arquivo de áudio nos formatos WAV ou PCM (a diferença entre arquivos WAV e PCM é que o WAV contém um cabeçalho com metadados sobre o áudio, enquanto o PCM armazena apenas os dados brutos de áudio sem qualquer estrutura adicional), sendo mono, anecoico e deve ter a informação de diretividade da fonte sonora. A [Figura 15](#) mostra

a interface associada a esse bloco, em que o usuário pode fazer *upload* do áudio mono e iniciar o processo de configuração.

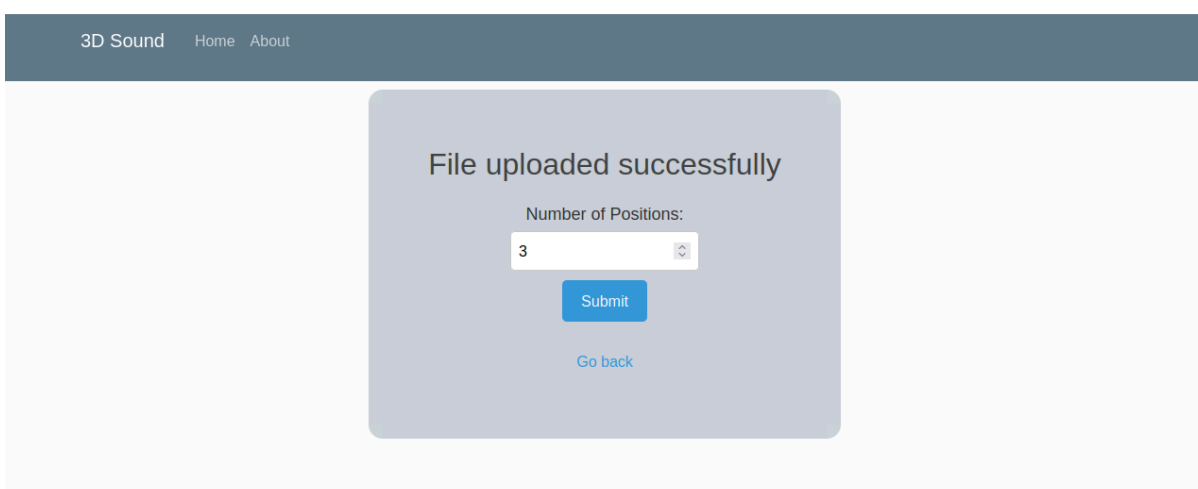
Figura 15 - Interface do bloco “Upload da entrada”.



Fonte : Autor.

O bloco “Quantidade de Posições” é responsável por receber e processar o arquivo de áudio selecionado no bloco anterior. Esse arquivo é lido e salvo na IDE com um nome específico, que varia conforme a extensão (por exemplo, “conteudo.wav” ou “conteudo.pcm”). Após o *upload* do arquivo, a interface permite ao usuário selecionar o número de posições de áudio para a saída 3D, para a criação de um vetor de posição que define a trajetória da fonte sonora no ambiente virtual, configurando assim os parâmetros necessários para o processamento espacial, como ilustrado na [Figura 16](#).

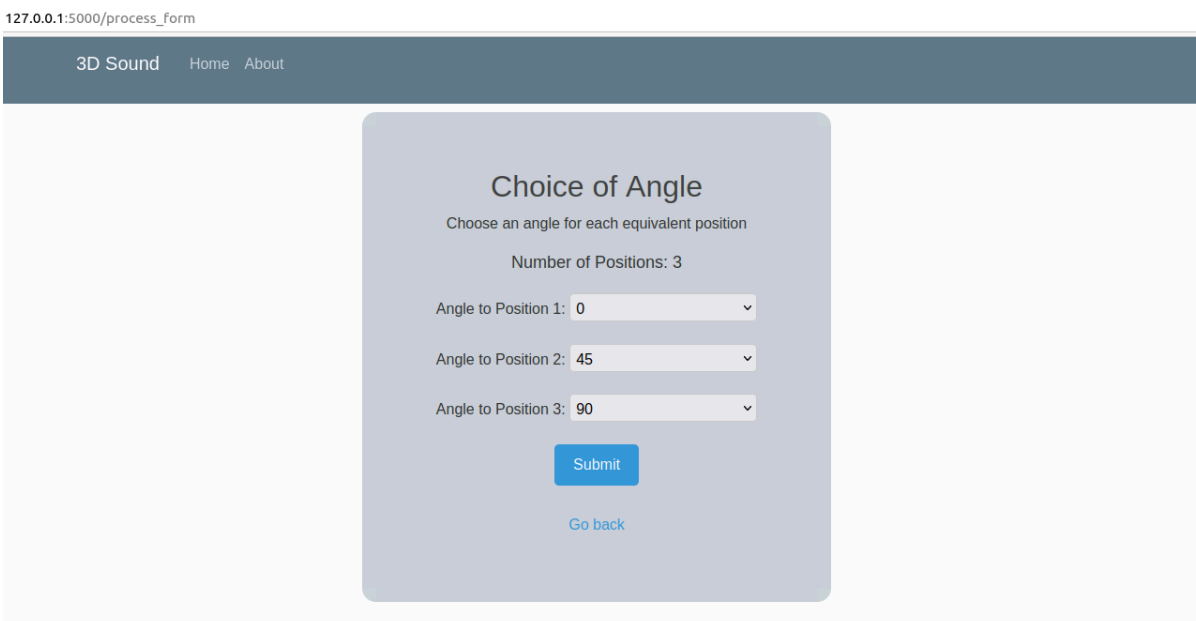
Figura 16 - Interface da bloco “Quantidades de posições”.



Fonte: Autor.

No bloco “Escolha dos ângulos para cada posição” recebe o número de posições definido anteriormente no bloco anterior. Com base nessa definição, ela permite que o usuário selecione os ângulos das direções desejados para cada posição que a fonte sonora terá na trajetória, personalizando assim a espacialização do som. Essa etapa de configuração dos ângulos é ilustrada na [Figura 17](#), permitindo que o usuário especifique as direções para a saída 3D, essenciais para o processamento de áudio espacial.

Figura 17 - Interface do bloco “Escolha dos ângulos para cada posição”.



127.0.0.1:5000/process_form

3D Sound Home About

Choice of Angle

Choose an angle for each equivalent position

Number of Positions: 3

Angle to Position 1: 0

Angle to Position 2: 45

Angle to Position 3: 90

Submit

[Go back](#)

Fonte: Autor.

No bloco “Arquivo tipo Wav” realiza a análise se o arquivo é do tipo Wav. Para arquivos no formato Wav, as informações, como a taxa de amostragem (F_s), o número de *bits* e a duração do áudio, são extraídas diretamente do cabeçalho do arquivo. Já para arquivos do tipo PCM, é solicitado ao usuário na interface do bloco “Obtém as informações do PCM” que informe a taxa de amostragem (F_s) e o número de *bits*, uma vez que esses dados não estão presentes no cabeçalho do arquivo. A interface correspondente é exibida na [Figura 18](#), detalhando essas configurações para cada tipo de arquivo.

Figura 18 - Interface do bloco “Obtém as informações do PCM”.

127.0.0.1:5000/process_form_angle

3D Sound Home About

Information about pcm file

Frame Rate:
44100

Bits:
16

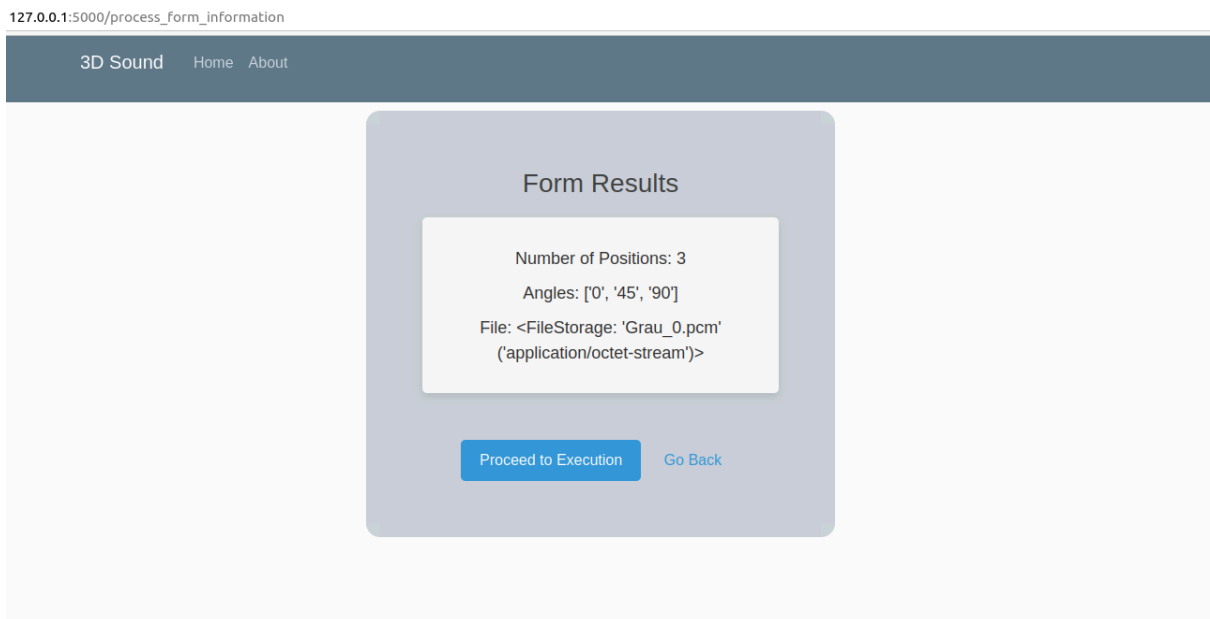
Submit

Go Back

Fonte : Autor.

No bloco “Processamento de informações” processa as informações obtidas sobre o tipo de arquivo definido na etapa anterior. Caso o arquivo seja do tipo PCM, a duração é calculada em segundos, já que essa informação não está disponível diretamente. Para arquivos Wav, a duração não precisa ser calculada, pois é extraída diretamente do cabeçalho. Após obtenção de todas as informações necessárias, um formulário de resultado (*Form Result*) é exibido, conforme na [Figura 19](#), onde são apresentadas todas as especificações reunidas nas etapas anteriores. Em seguida, o usuário pode iniciar o processamento do áudio de entrada com as configurações definidas, clicando no botão “*Proceed to Execution*”.

Figura 19 - Interface do bloco “Processamento de informações”.



Fonte: Autor.

No bloco “Aplicação” executa o processamento principalmente ao compilar e executar os códigos em linguagem C através do *script* em Python, conforme ilustrado na [Figura 20](#). Inicialmente, ocorre a compilação e execução do arquivo “*vetores_D.c*” usando o compilador GCC. Em seguida, o código “*main.c*” é compilado e executado. Após essa etapa, o arquivo de saída gerado é convertido de formato PCM para Wav, garantindo a compatibilidade para o *download*. O arquivo resultante estará disponível para o usuário no bloco “Realizar o *download*”, como mostrado na [Figura 21](#).

Figura 20 - Código em Python para a chamada da execução do processamento em C.

```
@app.route('/application')
def application():

    print("Executará o código em C adiante")

    import subprocess

    # Step 1: Compile the C code using a C compiler (e.g., GCC)
    c_code_file = "/home/linse/Documentos/Audio3D_Oficial/vetores_D.c"
    compiled_binary = "my_c_program" # Name of the compiled binary file

    compiler_command = ["gcc", c_code_file, "-o", compiled_binary]
    subprocess.run(compiler_command, check=True) # Check for compilation errors

    # Step 2: Execute the compiled binary
    execution_command = ["/." + compiled_binary]
    subprocess.run(execution_command)

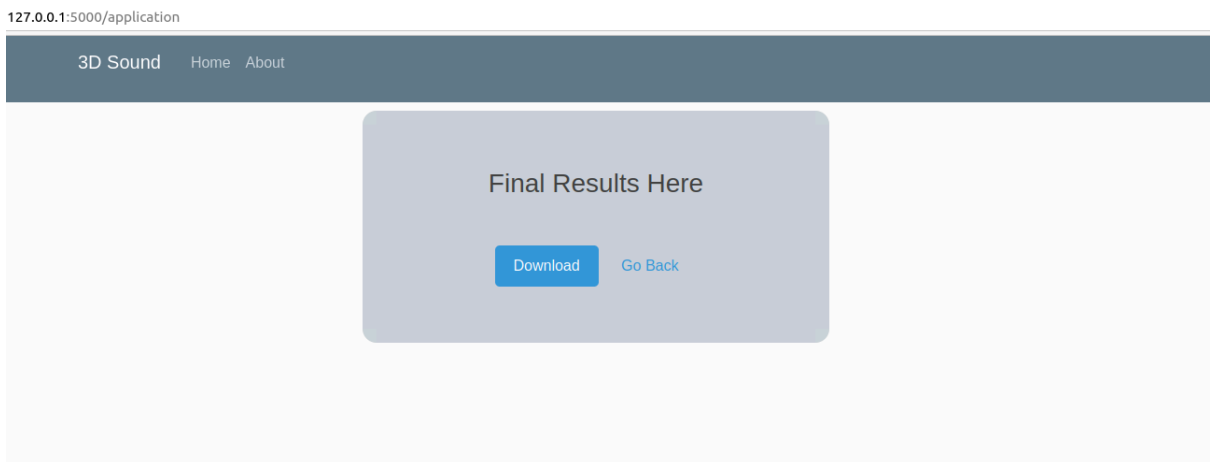
    # Step 1: Compile the C code using a C compiler (e.g., GCC)
    c_code_file = "/home/linse/Documentos/Audio3D_Oficial/main.c"
    compiled_binary = "my_c_program" # Name of the compiled binary file

    compiler_command = ["gcc", c_code_file, "-o", compiled_binary]
    subprocess.run(compiler_command, check=True) # Check for compilation errors

    # Step 2: Execute the compiled binary
    execution_command = ["/." + compiled_binary]
    subprocess.run(execution_command)
```

Fonte: Autor.

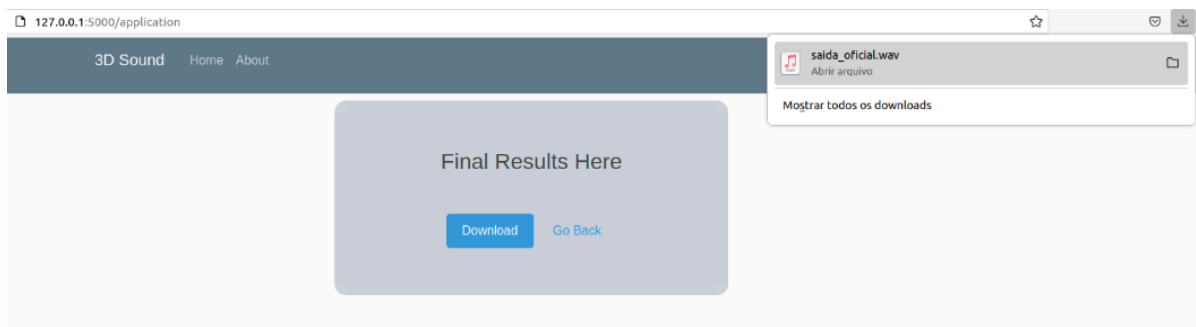
Figura 21 - Interface do bloco “Realizar o *download*”.



Fonte: Autor.

A “Saída áudio binaural” é o resultado ao usuário baixar o arquivo de saída final, denominado de "*saída_oficial.wav*". Esse arquivo contém o áudio processado em formato Wav, pronto para ser utilizado conforme as configurações definidas pelo usuário durante as etapas anteriores, obtendo a saída áudio binaural como ilustrado na [Figura 22](#).

Figura 22 - Interface da “Saída áudio binaural”.



Fonte: Autor.

3.4 DESENVOLVIMENTO EM PYTHON

3.4.1 Aquisição das HRTFs

Para iniciar o processo de espacialização, as HRIRs são adquiridas pela extração realizada do banco de HRTFs do projeto SADIE II, conforme os ângulos selecionados previamente. Neste projeto, utilizam-se HRTFs dos ângulos de 0° a 315° , com incremento de 45° , mantendo uma elevação constante de 0° , o que resulta em 8 HRTFs para cada canal (esquerdo e direito).

Python foi a ferramenta escolhida para facilitar o acesso e manipulação do banco de dados SADIE II, permitindo selecionar de maneira eficiente os ângulos e elevações desejadas e obter as HRTFs correspondentes. Essa etapa prepara o projeto para o processamento em áudio 3D, garantindo que as HRTFs estejam de acordo com as posições espaciais especificadas.

3.4.2 IDE

A IDE desenvolvida em Python usa o *framework* Flask para criar uma aplicação *web* dinâmica. O Flask facilita a construção e organização de rotas, que representam as URLs às quais a aplicação responde. Essas rotas mapeiam URLs para funções Python específicas, definindo como a aplicação reage a diferentes solicitações.

Por exemplo, ao acessar a rota `"/home"`, o Flask executa a função associada, que pode renderizar uma página HTML ou realizar operações específicas. A [Figura 23](#) mostra um trecho do código que exemplifica as rotas `"/home"` e `"/about"`, ilustrando como elas se conectam às respectivas funções. Essas funções podem lidar com dados enviados pelo usuário, acessar bases de dados, processar informações e retornar respostas, seja em formato de página HTML ou em outros formatos compatíveis com a *web*.

Esse sistema de rotas permite modularidade e organização no desenvolvimento, facilitando a expansão e manutenção do código. Ele também oferece flexibilidade para criar respostas interativas e dinâmicas, essenciais para a interface que envolve o processamento de áudio e personalização de parâmetros pelo usuário.

Figura 23 - Exemplo de rotas da IDE.

```
@app.route("/") #rotas para direcionar solicitações HTTP
@app.route("/home")
def home():
    return render_template('home.html')

@app.route("/about")
def about():
    return render_template('about.html', title='About')
```

Fonte : Autor.

3.5 DESENVOLVIMENTO EM C

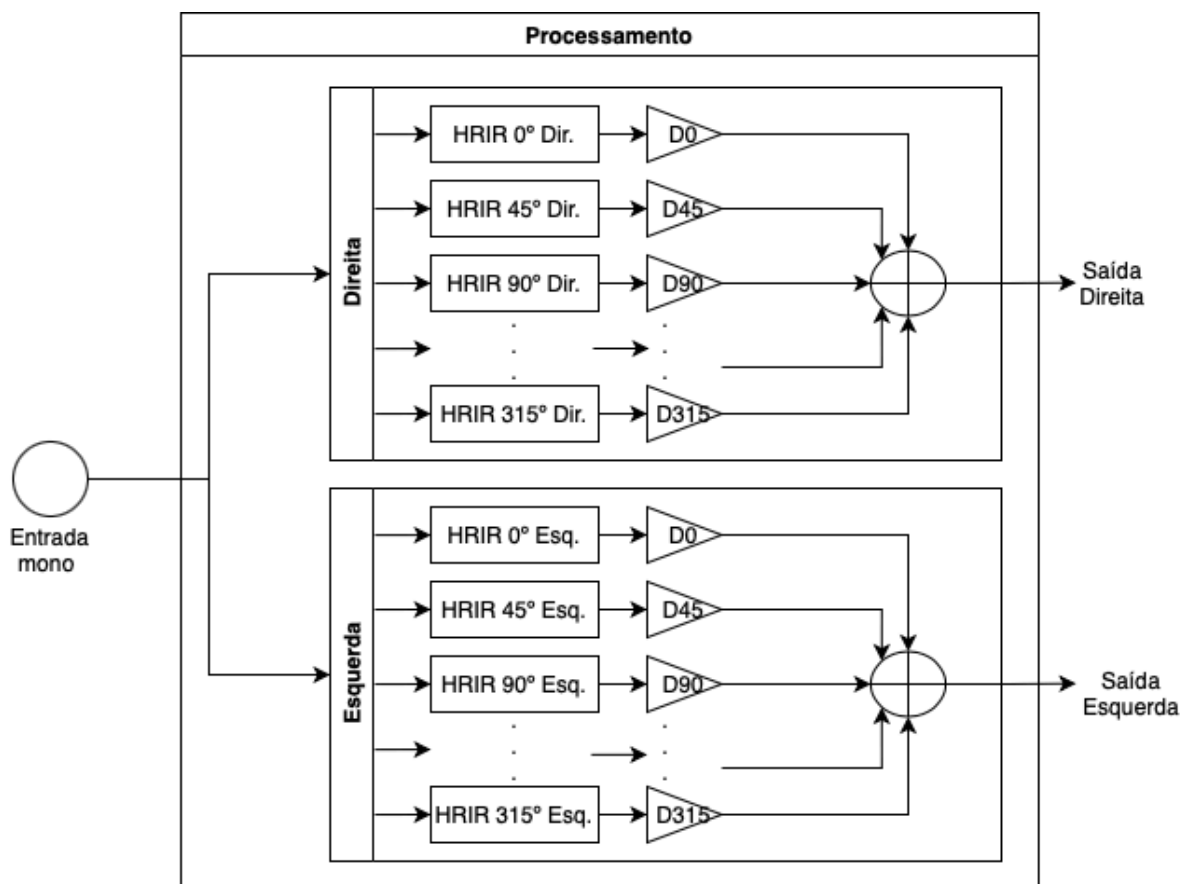
O desenvolvimento em C é essencial para a conversão do áudio mono em um áudio binauricular, dada a eficiência e controle direto que essa linguagem oferece para operação de processamento intensivo, como convoluções. O algoritmo, implementado no processamento, está estruturado de acordo com o diagrama de blocos apresentado na [Figura 24](#).

Neste diagrama, observa-se que a entrada de áudio mono passa por dois blocos principais, Direita e Esquerda, que representam as saídas estéreo. Cada bloco realiza a convolução com as HRIRs correspondentes aos ângulos específicos. Após essa etapa, o sinal convoluído é multiplicado pelos vetores de ganho específicos, chamados de

“vetores D”, os quais serão discutidos posteriormente. As saídas dessas multiplicações são então somadas, proporcionando a saída para cada canal de áudio.

As saídas processadas para cada canal são então somadas, gerando os sinais finais de áudio estéreo em 3D. Essa sequência de processamento, ilustrada no diagrama de blocos na [Figura 24](#), integra-se à interface desenvolvida para permitir a integração completa de conversão de áudio mono para 3D, como previamente mostrada no diagrama da [Figura 13](#).

Figura 24 - Diagrama de blocos do processamento.



Fonte: Autor.

3.5.1 Criação dos vetores das posições

O código responsável pela construção dos vetores, denominados “Vetores D” que representam as posições espaciais de áudio para aplicar as HRIRs das HRTFs correspondentes, considerando informações como taxa de amostragem (F_s), ângulos definidos e a duração do arquivo de entrada. Esse processo permite ativar ou desativar

posições específicas ao longo do tempo, conforme os parâmetros definidos pelo usuário. A seguir será detalhado o funcionamento:

3.5.1.1 *Coleta de Informações*

O código primeiramente recebe dados de entrada como F_s , os ângulos de direção definidos pelo usuário e a duração do arquivo. Essas informações determinam as características dos vetores de posição ao longo do tempo.

3.5.1.2 *Cálculo de Parcelas de Tempo*

A duração do áudio é dividida de acordo com o número de posição. Se o arquivo tiver 30 segundos e o usuário especificar 5 posições, cada posição ocupará um intervalo de 6 segundos. Esse cálculo permite definir o tempo de ativação de cada ângulo.

3.5.1.3 *Construção dos Vetores D*

Com base nas posições fornecidas, vetores específicos são criados para representar a ativação dos ângulos ao longo do tempo. Por exemplo, para o caso em que o usuário escolhe as posições [0, 45, 90, 45, 0] graus, os vetores são criados da seguinte maneira:

Para a posição 0 grau: O vetor D_0 terá valores 1 durante os primeiros 6 segundos e durante os últimos 6 segundos. Em todos os outros momentos, terá o valor 0. Como pode ser observado na [Figura 25](#).

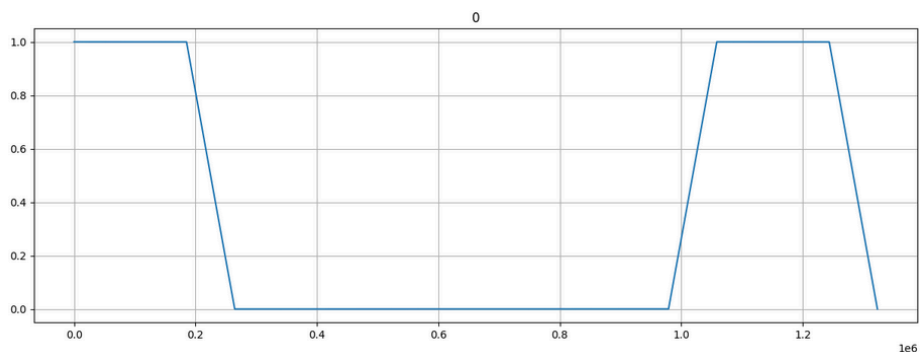
Para a posição 45 graus: O vetor D_{45} terá valores 0 durante os primeiros 6 segundos (0s - 6s), durante os próximos 6 segundos (6s - 12s) terá valor 1, já para os próximos 6 segundos (12s - 18s) será 0, para os seguintes 6 segundos (18s - 24s) o valor será 1. E para os últimos 6 segundos será 0. Como pode ser observado na [Figura 26](#).

Para a posição 90 graus: O vetor D_{90} só terá o valor 1 no terceiro estágio de tempo (12s - 18s) o restante será zero. Como pode ser observado na [Figura 27](#).

Para as posições restantes (135° , 180° , 225° , 270° e 315°): Os vetores correspondentes a essas posições terão todos os valores igual a zero, indicando que essas posições não estão ativas em nenhum momento.

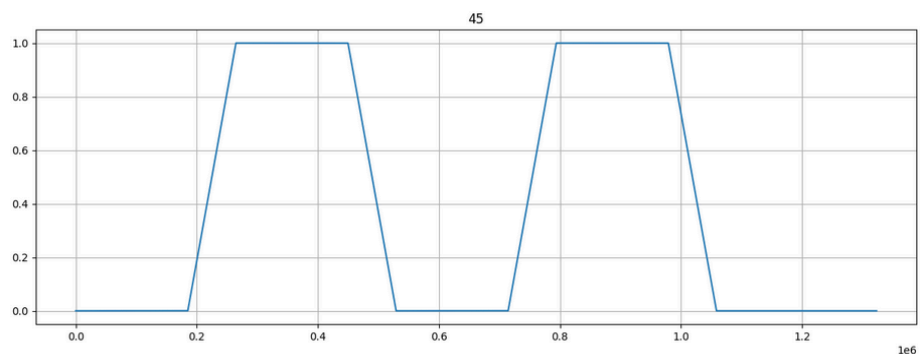
Para evitar mudanças abruptas entre posições, o código implementa transições suaves entre 0 e 1, aplicando uma rampa de 30% do intervalo de tempo. O uso da rampa é para não haver uma atenuação no ganho, somando em todo instante do tempo os ganhos aplicados, resulta em 1. Isso resulta em uma troca gradual das posições, o que melhora a percepção espacial do áudio sem ruídos de transição brusca. Na [Figura 28](#), é possível observar todos os vetores que foram utilizados no exemplo. Percebe-se que o vetor D_0 começa a reduzir antes dos 6 segundos (mais especificamente em 70% de 6s) e nesse instante o vetor D_90 começa a aumentar atingindo amplitude 1 exatamente em 6s. A rampa de transição ilustra como os vetores ativam gradualmente as posições de áudio ao longo do tempo, garantindo uma experiência auditiva mais natural e suave.

Figura 25 - Vetor D_0 do ângulo 0° .



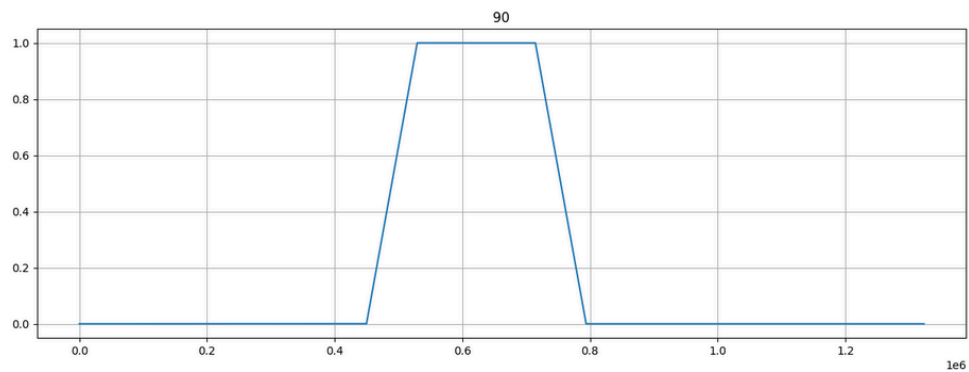
Fonte: Autor.

Figura 26 - Vetor D_45 do ângulo 45° .



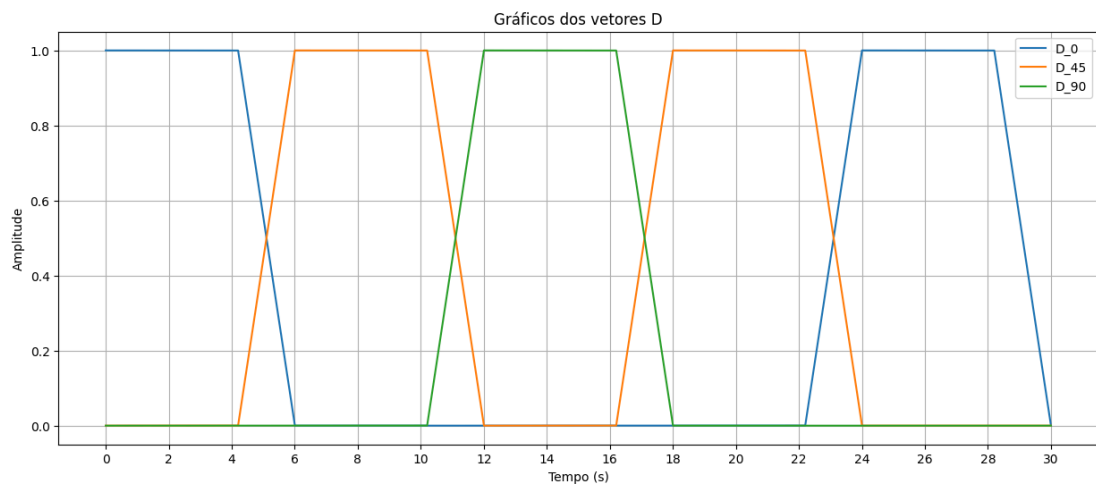
Fonte: Autor.

Figura 27 - Vetor D_90 do ângulo 90°.



Fonte: Autor.

Figura 28 - Todos os vetores utilizados no exemplo.



Fonte: Autor.

3.5.2 Processamento

O código do processamento principal (*main*) é a base do processamento que converte o áudio mono anecoico em uma saída binaural 3D. Esse processamento é realizado através de convoluções do áudio de entrada com as HRIRs específicas para cada ângulo e da multiplicação do resultado pelos vetores de direção para cada posição, resultando na geração do áudio tridimensional. As principais partes do código são:

3.5.2.1 Função *main*

A função principal do programa é responsável por configurar o processamento. Obtém informações sobre os arquivos de áudio (como nome, bits e tipo do arquivo) a partir de arquivos de texto gerados pela interface em Python.

Chama a função “*openHRTFs*” para carregar as HRIRs das HRTFs correspondentes dos ângulos especificados.

Chama a função “*openFile*” para iniciar o processamento do áudio de entrada, realizando as convoluções com as HRIRs e gerando o áudio de saída em formato 3D.

3.5.2.2 Função *openHRTFs*

Esta função é usada para abrir arquivos de texto contendo respostas ao impulso (HRIRs) e armazená-las em vetores. Recebe como argumentos os ângulos de elevação e azimute para determinar o nome do arquivo a ser aberto.

Lê as HRIRs de cada canal (esquerdo e direito) e armazena as respostas de impulso em vetores correspondentes para cada canal. Esses vetores serão usados posteriormente na convolução com o áudio de entrada.

3.5.2.3 Função *openFile*

Esta função processa o áudio de entrada (Wav ou PCM) amostra por amostra. Recebe como argumentos o nome do arquivo Wav ou PCM, os bits do arquivo, o tipo de áudio e as *arrays* contendo respostas ao impulso das HRTFs.

Para cada amostra do arquivo de entrada, chama a função de convolução (função *conv*) para realizar a convolução com as HRIRs correspondentes, aplicando a resposta ao impulso para criar uma saída binaural espacial.

A saída da convolução é multiplicada pela amostra correspondente dos vetores das posições (Vetores D). Finalmente, o resultado dessas multiplicações são somados gerando o arquivo de saída PCM (“*saida_oficial.pcm*”).

3.5.2.4 *Função conv*

Essa função realiza a convolução do áudio de entrada com as HRIRs. Recebendo como argumentos as HRIRs pré-carregadas e o vetor de amostras de áudio. Executando a convolução para gerar uma nova amostra de saída, que simula a propagação espacial do som. Retornando o resultado da convolução para ser multiplicado pelo vetor de direção D na função “*openFiles*”.

3.5.2.5 *Resumo do Fluxo de Processamento*

O main configura o processo e carrega as HRIRs. O “*openFile*” lê o áudio de entrada, amostra por amostra, e convolui cada uma com as HRIRs através da função “*conv*”. Após a convolução, cada amostra é multiplicada pelo vetor D , e o resultado final é acumulado no arquivo de saída, que é convertido para estéreo 3D.

Esse fluxo possibilita a geração de um áudio que simula um espaço tridimensional, adaptando o som de acordo com a direção especificada para cada amostra.

4 RESULTADOS

Os testes para o áudio 3D foram realizados de forma qualitativa, foram usados diversos áudios monos PCMs e Wavs. Foram experimentadas duas abordagens principais para a avaliação.

A primeira, os ângulos das HRTFs foram selecionados em ordem crescente, proporcionando uma trajetória previsível das posições ao longo do áudio. Com este método, se teve grande clareza das posições e percebia a movimentação do áudio de uma forma sutil. No caso do áudio de uma pessoa falando, havia a sensação da movimentação da pessoa conforme as posições. E o outro teste, era com posições aleatórias, nesse modo perdia a fluidez da movimentação, no entanto ainda era de fácil percepção a mudança de posição. Esses testes foram realizados por um pequeno grupo do laboratório Linse da UFSC, utilizando os equipamentos disponíveis no local.

Os resultados são encontrados em [26] junto com suas respectivas entradas. Onde é possível observar quatro resultados, sendo o "*Caminhada_3D.wav*" o áudio de uma caminhada que passa da posição 0° até a posição 315° em um passo de 45°. No "*Musica.wav*" o áudio é música instrumental que passa por diversas posições, sendo a ordem das posições: 0°, 180°, 45°, 225°, 270°, 90°, 315°, 90°, 270° e 0°. No "*UnicoTom_3D*" é o tom do seno em 500 Hz nas seguintes posições: 0°, 90°, 180°, 270°. E por último no "*VozTodosAngulos_3D.wav*" é uma voz falando as posições em que se encontra o áudio, dando uma volta completa de 0° a 360° em um passo de 45°.

Esses testes evidenciam que o algoritmo desenvolvido possibilita uma percepção das direções do áudio, seja em sequências organizadas ou aleatórias, com boa capacidade de percepção da movimentação espacial.

4.1 ANÁLISES

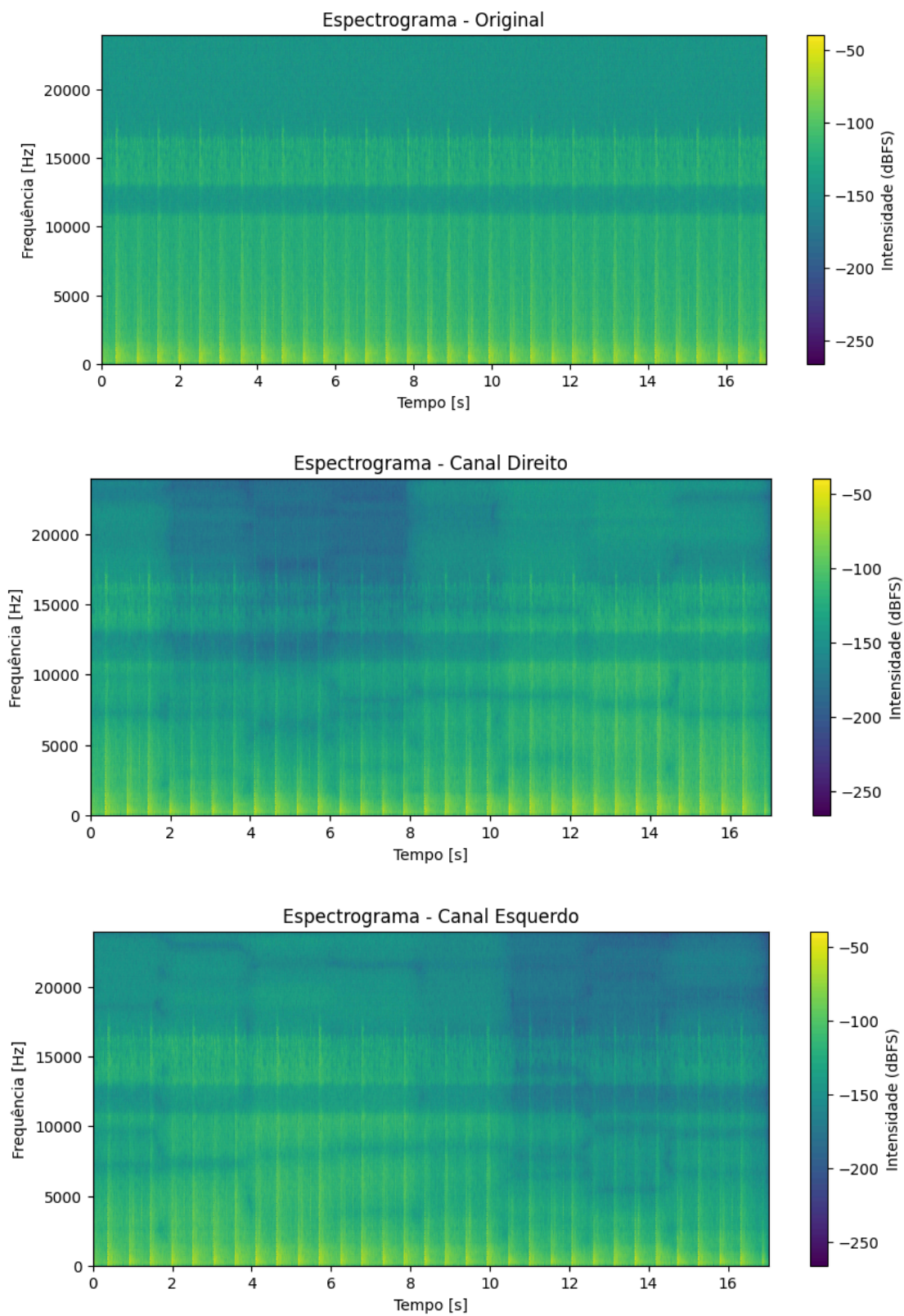
Para tornar os resultados visíveis e permitir uma análise quantitativa da espacialização do áudio, foram gerados espectrograma do áudio processado para cada canal. Na [Figura 29](#), são apresentados os espectrogramas original da caminhada (a), do canal esquerdo (a) e do canal direito (b) dos resultados do áudio 1, nos espectrogramas são mostradas faixas de intensidades variadas ao decorrer do tempo, evidenciando as mudanças nas posições angulares ao longo do tempo.

Realizando a análise das mudanças de intensidade ao decorrer do tempo, consegue-se notar que no intervalo de tempo de 2 a 8 segundos mostram maior intensidade no lado esquerdo (a) e no intervalo entre 10 a 16 segundos tem maior intensidade no lado direito (b). Isso se dá ao fato do intervalo de 2 a 8 segundos estarem localizadas no lado esquerdo da cabeça. Essa maior intensidade no canal esquerdo reflete a proximidade da fonte de som com a orelha esquerda nesses ângulos.

De forma análoga, no intervalo de 10 a 16 segundos corresponde às posições que se encontram no lado da orelha direita. E no intervalo de 0 a 2 segundos, correspondem à posição de 0° e o intervalo de 8 a 10 segundos corresponde à posição de 180° e é perceptível que as intensidades são similares do canal esquerdo com o canal direito, devido estarem na frente e atrás da cabeça.

Esses espectrogramas proporcionam uma visualização da mudança espacial do áudio e da intensidade relativa em cada orelha, facilitando uma análise objetiva da distribuição espacial e da direção da fonte sonora ao longo do tempo. Essa visualização ajuda a validar a eficácia do modelo de espacialização, mostrando como a percepção do áudio 3D é ajustada conforme as posições angulares configuradas.

Figura 29 - Espectrogramas original (a), do canal esquerdo (a), do canal direito (b) dos resultados do áudio da caminhada.



Fonte: Autor.

Para demonstrar os resultados e a eficácia do processamento de espacialização, foi utilizado um tom senoidal de 100 Hz como áudio de entrada. Na [Figura 30](#), é possível observar uma comparação entre o sinal de entrada e as saídas processadas para cada canal (esquerdo e direito) em uma posição angular de 90°.

Comparando as saídas dos canais esquerdo e direito com o sinal de entrada é perceptível a diferença entre os sinais, percebe-se que ambas as saídas têm amplitude reduzida. No entanto, o canal direito apresenta uma intensidade menor do que o canal esquerdo, refletindo a posição de 90°, onde a fonte está mais próxima a orelha esquerda. Há também um atraso perceptível entre os sinais dos canais esquerdo e direito. Esse atraso temporal corresponde à diferença de tempo que o som leva para chegar a cada orelha devido à posição espacial da fonte.

Essa diferença de intensidade, caracterizada como ILD, que representa a diferença da intensidade entre os canais direitos e esquerdos, pode ser calculada pela equação 2:

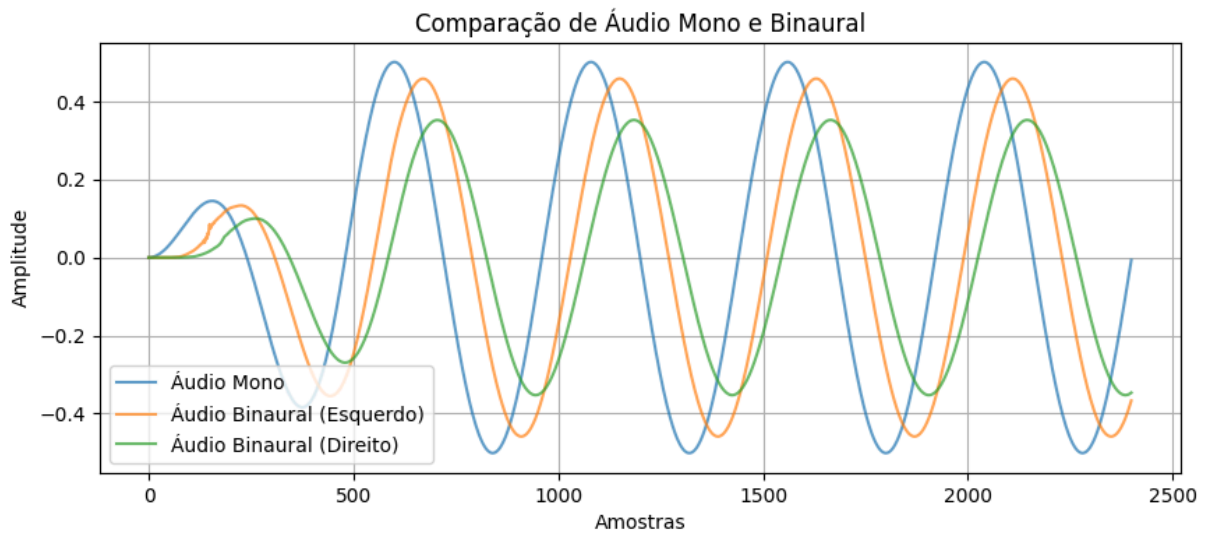
$$ILD_{sin} = 20 \cdot \log_{10} \left(\frac{A_{direita}}{A_{esquerda}} \right) \quad (2)$$

Onde A é o valor médio quadrático da amplitude para cada canal, resultando a diferença em dB do canal da direita com o esquerdo. Para o sinal processado, o cálculo da ILD resultou em -2,32 dB, indicando que o canal esquerdo tem maior intensidade.

E calculando a diferença temporal dos sinais, resultou em uma ITD de 0,729 ms. Esse valor representa que o tempo do som levou mais para chegar na orelha direita em relação ao esquerdo, caracterizando a percepção espacial de um som que se encontra à esquerda do ouvinte.

Essas medidas de ILD e ITD validam a implementação do modelo de áudio 3D, mostrando como a percepção espacial é manipulada para criar a sensação realista de direção e distância com base nas diferenças de intensidade e de tempo entre os canais.

Figura 30 - Comparação da entrada e saída de cada canal de um sinal senoidal de 100 Hz com a fonte na posição 90°.



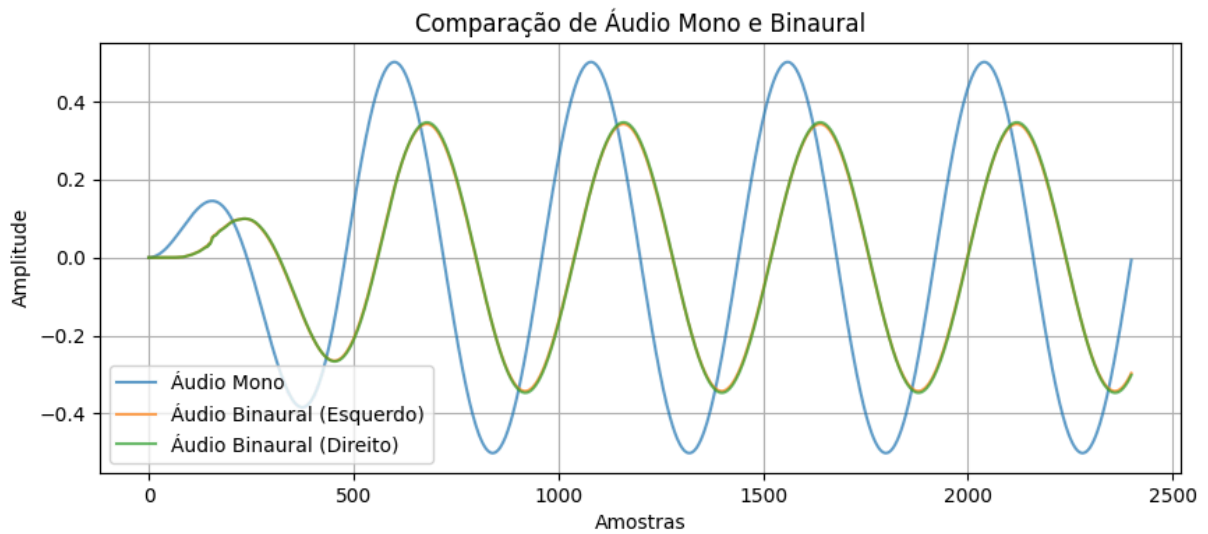
Quando o mesmo sinal de entrada senoidal de 100 Hz é processado na posição 180° (atrás do ouvinte), os resultados são apresentados na [Figura 31](#) mostram sinais de saída no canal esquerdo e direito que são muito similares. No entanto, ainda existem pequenas diferenças entre os dois canais, o que caracteriza a percepção de uma fonte sonora posicionada diretamente atrás.

A diferença de intensidade (ILD) entre os canais direito e esquerdo é pequena, com um valor de 0,11 dB. Essa diferença quase imperceptível indica que o som chega com praticamente a mesma intensidade em ambas as orelhas, o que é esperado para uma fonte sonora posicionada atrás do ouvinte.

A diferença de tempo (ITD) é também mínima, com um valor de 0,021 ms. Esse pequeno atraso entre os canais indica que o som atinge ambas as orelhas quase simultaneamente, reforçando a percepção de que a fonte está em uma posição central (posterior).

Essas características, com valores muito baixos de ILD e ITD, contribuem para a sensação de que o som está vindo de uma posição diretamente atrás da orelha. Isso confirma que o modelo de áudio 3D consegue reproduzir diferenças sutis nas intensidades e nos tempos de chegada para criar uma percepção espacial precisa.

Figura 31 - Comparação da entrada e saída de cada canal de um sinal senoidal de 100 Hz com a fonte na posição 180°.



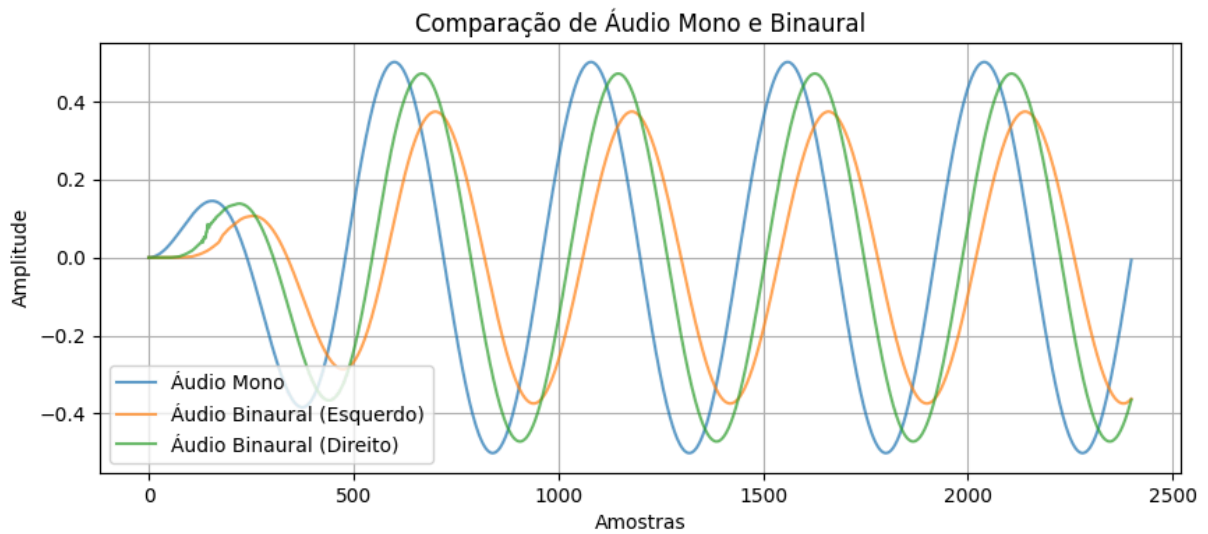
Quando o sinal de entrada senoidal de 100 Hz é processado na posição 270° (diretamente à direita do ouvinte), o resultado na [Figura 32](#) apresenta características semelhantes ao processamento na posição de 90° (diretamente à esquerda), mas com uma diferença importante: o canal direito agora apresenta uma amplitude maior do que o canal esquerdo.

A ILD entre os canais direito e esquerdo é de 2,05 dB. Nesse caso, o ILD é positivo indica que o som está mais intenso no canal direito, o que é consistente com a posição da fonte sonora diretamente à direita do ouvinte.

O ITD entre os canais é -0,687 ms, indicando que o som chega mais cedo no canal direito em comparação com o esquerdo. Esse atraso negativo reflete o fato de que o som proveniente atinge a orelha direita antes da esquerda.

Esses valores de ILD e ITD são coerentes com a localização da fonte sonora à direita, onde a orelha direita capta o som mais cedo com maior intensidade. A inversão das características em relação à posição de 90° reforça a eficácia do modelo em representar espacialmente a posição de 270°. Combinado esses parâmetros, o sistema consegue fornecer ao ouvinte uma percepção clara e natural de que o som vem da direita.

Figura 32 - Comparação da entrada e saída de cada canal de um sinal senoidal de 100 Hz com a fonte na posição 270°.



Foram realizados os processamentos dos sinais de 100 Hz, 500 Hz e 2000 Hz para as posições de 0° até a posição 315° em um passo de 45°. E foi obtida o ILD em dB, a diferença de amplitude mono com o canal da esquerda em dB, a diferença de amplitude mono com o canal da direita em dB, o ITD, a diferença do tempo entre a entrada mono e o canal direito e a diferença do tempo entre a entrada mono com o canal da esquerdo.

Na Tabela 2 se encontra os resultado do sinal de 100 Hz, analisando os valores consegue-se tirar algumas conclusões:

Em 0° o ILD é 0,14 dB, logo a intensidade que chega em ambos os orelhas é muito próxima, como o ITD que é de 0,021 ms mostrando também que a diferença de tempo da chegada do som nas orelhas é muito próxima, isso acaba caracterizando a posição frontal.

Enquanto em 90° percebe-se que é onde teve a maior módulo de ILD de -2,42 dB, analisando a diferença de amplitude mono com o canal esquerdo e direito, dá para analisar que o canal da direita teve uma maior atenuação com -3,25 dB que o lado esquerdo com -0,83 dB, logo o som chegará mais intenso no orelhas esquerdo. E o módulo de ITD também foi o maior com 0,729 ms, sendo o som do lado esquerdo chegando antes do lado direito.

Observando o resultado em 270°, é notável o resultado oposto de 90°. Sendo o canal direito, menos atenuado e que sofre o menor atraso, fazendo com o som esteja mais próximo da orelha direita.

Tabela 2 – Resultado do sinal senoidal de 100 Hz.

Ângulo das posições da fonte	ILD (dB)	Diferença de amp. mono vs esquerdo (dB)	Diferença de amp. mono vs direito (dB)	ITD (ms)	Diferença do tempo entre mono e direito (ms)	Diferença do tempo entre mono e esquerdo (ms)
0°	0,14	-1,61	-1,48	0,021	1,812	1,792
45°	-0,20	-1,31	-1,50	0,396	1,979	1,583
90°	-2,42	-0,83	-3,25	0,729	2,188	1,458
135°	-1,87	-1,72	-3,59	0,542	2,021	1,479
180°	0,10	-3,40	-3,30	0,021	1,667	1,646
225°	1,85	-3,30	-1,44	-0,500	1,417	1,917
270°	2,14	-2,71	-0,58	-0,688	1,396	2,083
315°	0,43	-1,50	-1,07	-0,375	1,562	1,938

Fonte: Autor.

Na Tabela 3 se encontra os resultado do sinal de 500 Hz, analisando os valores consegue-se tirar algumas conclusões:

Que as características são similares já descritas do sinal de 100 Hz, mas consegue-se notar certas diferenças, como um módulo de ILD geral maior, tendo casos como por exemplo 90°, onde há um aumento na amplitude em relação a entrada mono, tendo uma diferença de 1,78 dB positivo, também se teve diferenças de tempo menores, mas ITD na maior parte maiores.

Tabela 3 – Resultado do sinal senoidal de 500 Hz.

Ângulo das posições da fonte	ILD (dB)	Diferença de amp. mono vs esquerdo (dB)	Diferença de amp. mono vs direito (dB)	ITD (ms)	Diferença do tempo entre mono e direito (ms)	Diferença do tempo entre mono e esquerdo (ms)
0°	-0,50	-2,57	-3,08	-0,021	0,250	0,271
45°	-6,14	0,99	-5,15	0,562	0,583	0,021
90°	-5,72	1,78	-3,94	0,896	0,833	-0,062
135°	-6,05	0,36	-5,69	0,625	0,604	-0,021
180°	-0,26	-3,38	-3,63	0,021	0,167	0,146
225°	6,29	-5,74	0,54	-0,625	-0,104	0,521
270°	5,81	-3,90	1,91	-0,896	-0,146	0,750
315°	5,86	-5,05	0,81	-0,562	0,000	0,562

Fonte: Autor.

Na Tabela 4 se encontram os resultados do sinal de 2000 Hz, analisando os valores consegue-se mostrar com mais evidência o que já foi citado na análise de 100 Hz e 500 Hz da tabela 1 e 2.

Tabela 4 – Resultado do sinal senoidal de 2000 Hz.

Ângulo das posições da fonte	ILD (dB)	Diferença de amp. mono vs esquerdo (dB)	Diferença de amp. mono vs direito (dB)	ITD (ms)	Diferença do tempo entre mono e direito (ms)	Diferença do tempo entre mono e esquerdo (ms)
0°	-1,04	-0,02	-1,07	-0,021	0,000	0,021
45°	-8,66	1,39	-7,27	0,396	0,229	-0,167
90°	-9,11	-0,02	-9,13	0,729	0,500	-0,229
135°	-8,47	-0,11	-8,57	0,396	0,208	-0,188
180°	-1,75	-3,83	-5,58	0,021	-0,062	-0,083
225°	7,28	-8,03	-0,75	-0,396	-0,271	0,125
270°	10,94	-10,05	0,89	-0,771	-0,333	0,438
315°	7,57	-5,16	2,42	-0,438	-0,208	0,229

Fonte: Autor.

A análise realizada mostra a relação entre a percepção sonora e os parâmetros acústicos ILD (Diferença de Nível Interaural) e ITD (Diferença de Tempo Interaural).

Conforme apresentado em [30] mostra-se que o ILD se torna mais perceptível para frequências altas, enquanto o ITD predomina em baixas frequências. A análise dos resultados apresentados nas tabelas 2 a 4 mostra que o ILD aumenta à medida que a frequência também aumenta.

De acordo com [31], a Diferença Mínima Perceptível (JND) para ILD deve ser maior que 0,5 dB para permitir a identificação da direção da fonte sonora. Na tabela 2, para uma senoide de 100 Hz, o intervalo de 45° para 90° apresentou uma diferença de ILD de -2,22 dB, permitindo identificar a mudança na direção da fonte sonora. Contudo, no intervalo de 0° para 45°, a diferença de ILD foi de -0,34 dB, insuficiente para percepção pela ILD, sendo a mudança perceptível pela ITD, com uma diferença de 0,375 ms.

Observa-se da tabela 3, para uma senoide de 500 Hz, a diferença de ILD de 0° para 45° foi de -5,64 dB, permitindo identificar a mudança na direção da fonte sonora. No entanto, para o intervalo de 45° a 90°, a diferença de ILD foi de 0,42 dB, abaixo do limiar de percepção. Nesse caso, a diferença da ITD foi de 0,334 ms, permitindo identificar a mudança da posição.

Finalmente, na tabela 4 para uma senoide de 2000 Hz a diferença de ILD de 45° para 90° foi de 0,45 dB, sendo a única abaixo de 0,5 dB, não sendo perceptível identificar a mudança na direção da fonte sonora pela ILD. Contudo, a diferença da ITD foi de 0,333 ms, permitindo a distinção na mudança de posição.

Com base nos resultados, percebe-se que o aumento da frequência resulta no aumento das diferenças de ILD e na diminuição das diferenças de ITD. Tal resultado reforça a importância de utilizar ambos os parâmetros para a localização da fonte sonora.

5 CONCLUSÃO

A utilização das HRIRs das HRTFs correspondentes como ferramenta para o processamento de áudio 3D mostrou-se altamente eficaz, permitindo uma espacialização realista e configurável de fontes sonoras. Este método supera a complexidade e limitações da gravação biauricular tradicional, oferecendo flexibilidade para ajustar a posição de qualquer som em um espaço virtual de forma prática e precisa. Os resultados obtidos indicam que o processamento com HRIRs gera uma sensação clara de direção, com variação perceptíveis de intensidade (ILD) e tempo (ITD) conforme a posição da fonte sonora, aprimorando a experiência imersiva.

O projeto apresenta, além de oferecer alta qualidade de processamento, destaca-se pela facilidade de uso. A interface foi desenvolvida para simplificar a manipulação de áudio 3D, tornando acessível para usuários de diferentes níveis de experiência a configuração de posições sonoras sem a necessidade de conhecimento de técnicas em HRTFs ou programação. Este avanço na usabilidade, somado à fidelidade espacial proporcionada, amplia as possibilidades de aplicação do áudio 3D em áreas como realidade aumentada, entretenimento interativo e simulações acústicas, consolidando o uso de HRTFs como uma solução prática e acessível para produção de áudio imersivo.

5.1 TRABALHOS FUTUROS

Para futuros desenvolvimento, uma área promissora é a adição de mais posições pré-estabelecidas para aumentar a versatilidade do software. Outra melhoria seria permitir a personalização do tempo em cada posição, o que agregaria complexidade ao controle dos vetores das posições, mas tornaria o software mais próximo de um editor de mídia completa, dispensando a necessidade de programas externos para tais edições.

Além disso, explorar o uso de inteligência artificial para a separação de fontes sonoras seria um avanço significativo. Por exemplo, em uma faixa de música, a IA poderia isolar a voz, o violão e o piano, permitindo que esses elementos sejam posicionados em diferentes locais no espaço 3D. Essa técnica acrescentaria um dinamismo inovador à música, criando uma experiência mais imersiva e interativa.

REFERÊNCIAS

- [1] ROAD, Curtis. *The computer music tutorial*. Cambridge: MIT Press, 1996.
- [2] SOLLITO, André. Mudou de fase: mercado de games já fatura mais que o de cinema. *Veja*, 2023, n. 2860, 22 set 2023. Disponível em: <https://veja.abril.com.br/tecnologia/mudou-de-fase-mercado-de-games-ja-fatura-mais-que-o-d-e-cinema>. Acesso em: 1 out. 2024.
- [3] SPOTIFY. *Spotify company info* [online]. Disponível em: <https://newsroom.spotify.com/company-info/> Acesso em: 1 out 2024.
- [4] BEGAULT, Durand R.; TREJO, Leonard J. *3-D sound for virtual reality and multimedia*. Moffett Field: NASA Ames Research Center, 2000. (Technical Memorandum, NASA/TM-2000-209606; IH-010; NAS 1.15:209606). Disponível em: <https://ntrs.nasa.gov/citations/20010044352>. Acesso em: 1 nov. 2024.
- [5] AREVALO, Felipe Leonel Grijalva. *Manifold learning for spatial audio rendering: aprendizado de variedades para a síntese de áudio espacial*. 2018. Tese (Doutorado em Engenharia Elétrica – Área de Engenharia de Computação) – Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, 2018. Disponível em: <https://repositorio.unicamp.br/acervo/detalhe/1045122>. Acesso: 22 nov. 2024.
- [6] SOUZA, Gabriel Henrique Montesião de. *Auralização de fontes sonoras móveis usando HRTFs*. 2010. Dissertação (Mestrado em Ciências) – Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2011. Disponível em: https://www.teses.usp.br/teses/disponiveis/45/45134/tde-19092011-213316/publico/dissertacao_ghms.pdf. Acesso em: 22 nov. 2024.
- [7] SETTI, Fernanda Cardoso. *Síntese de som tridimensional*. 2015. Trabalho de Conclusão de Curso (Graduação em Engenharia de Controle e Automação) – Universidade de Caxias do Sul, Caxias do Sul, 2015. Disponível em: <https://repositorio.uces.br/xmlui/bitstream/handle/11338/2255/TCC%20Fernanda%20Cardoso%20Setti.pdf?sequence=1&isAllowed=y>. Acesso em: 22 nov. 2024.
- [8] RUMSEY, Francis. *Spatial audio*. Oxford: Focal Press, 2001.
- [9] BLAUERT, Jens. *Spatial hearing: the psychophysics of human sound localization*. Revised edition. Cambridge: The MIT Press, 1996. DOI: <https://doi.org/10.7551/mitpress/6391.001.0001>. ISBN 9780262268684.
- [10] YOST, William A. *Fundamentals of Hearing: An Introduction*. San Diego: Academic Press, 2007.
- [11] CHEUNG, W.-S.; PARK, Y.-H.; RUN, H. -J. Measurement of head vibration and its error analysis. In: INTER.NOISE 2000, The 29th International Congress and Exhibition on Noise Control Engineering, 27-30 ago 2000, Nice, França. Anais [...]. Nice: Conforg, 2000. Disponível em:

<http://www.conforg.fr/internoise2000/cdrom/data/articles/000166.pdf> Acesso em: 1 nov. 2024.

[12] PAUSCH, Florian; FELS, Janina. Localization performance in a binaural real-time auralization system extended to research hearing aids. *Trends in Hearing*, v.24, p. 1-18, 23 abr. 2020. DOI: 10.1177/2331216520908704. Disponível em: https://www.researchgate.net/figure/Definition-of-the-Used-Head-Related-Spherical-Coordinate-System-With-Azimuth-Angles-ph-and_fig1_340897099. Acesso em: 1 nov. 2024.

[13] BREEBART, Jeroen; FALLER, Christof. *Spatial audio processing: MPEG surround and other applications*. Chichester: John Wiley & Sons, 2007.

[14] TORTORA, G. J; DERRICKSON, B. **Princípios de Anatomia e Fisiologia**; Rio de Janeiro: Guanabara Koogan, 2016.

[15] THIEMANN, J.; VAN DE PAR, S. A multiple model high-resolution head-related impulse response database for aided and unaided ears. *EURASIP Journal on Advances in Signal Processing*, v. 2019, n. 9, 2019. DOI: <https://doi.org/10.1186/s13634-019-0604-x> . Disponível em: <https://asp-urasipjournals.springeropen.com/articles/10.1186/s13634-019-0604-x> Acesso em: 1 nov. 2024.

[16] GARDNER, Bill; MARTIN, Keith. *HRTF measurements of a KEMAR dummy-head microphone*. Cambridge: MIT Media Lab, 1994. (Technical Report, 280). Disponível em: <https://sound.media.mit.edu/resources/KEMAR/hrtfdoc.txt>. Acesso em: 1 nov. 2024.

[17] LISTEN HRTF DATABASE. Introduction. Disponível em: http://recherche.ircam.fr/equipes/salles/listen/system_protocol.html. Acesso em: 1 nov. 2024.

[18] CORDIS. FP6-IST Programme. Disponível em: <https://cordis.europa.eu/programme/id/FP6-IST>. Acesso em: 1 nov. 2024.

[19] HARMAN. Defining the Standart: The Science Behind AKG Reference Response Studio Headphones. Disponível em: <https://pro.harman.com/insights/akg/defining-the-standard-the-science-behind-akg-reference-response-studio-headphones/>. Acesso em: 1 nov. 2024.

[20] UNIVERSITY OF YORK. SADIE II Database. Disponível em: <https://www.york.ac.uk/sadie-project/database.html>. Acesso em: 1 nov. 2024.

[21] ARMSTRONG, Cal; THRESH, Lewis; MURPHY, Damian; KEARNEY, Gavin. A Perceptual Evaluation of Individual and Non-Individual HRTFs: A Case Study of the SADIE II Database. *Appl. Sci.*, v. 8, n. 11, p. 2029, 2018. Disponível em: <https://www.mdpi.com/2076-3417/8/11/2029>. Acesso em: 1 nov. 2024.

[22] DUCKETT, John. *HTML & CSS: Design and Build Websites*. John Wiley & Sons, INC., 201. Disponível em: <https://wtf.tw/ref/duckett.pdf>. Acesso em: 5 nov. 2024.

- [23] BASEL, Krzysztof. How Did Python Become the Language of Choice for Data Science?. Netguru Blog, 16 jun. 2023. Disponível em: <https://www.netguru.com/blog/python-for-data-science>. Acesso em: 5 nov. 2023
- [24] VAN ROSSUM, Guido. Python Tutorial. Technical Report CS-R9526, Centrum voor Wiskunde en Information (CWI), Amsterdam, Maio 1995.
- [25] NUMPY. NumPy Documentation. Disponível em: <https://numpy.org/doc/>. Acesso em: 5 nov. 2024.
- [26] SCIPY. SciPy Documentation. Disponível em: <https://docs.scipy.org/doc/>. Acesso em: 5 nov. 2024.
- [27] PALETS PROJECTS. 2024. Flask [Software]. Disponível em: <https://flask.palletsprojects.com/>. Acesso em: 5 nov. 2024.
- [28] KERNIGHAN, Brian; RITCHIE, Dennis. The C Programming Language. Prentice Hall, 1988.
- [29] PEREIRA, Luis. Resultados Áudio 3D, Google Drive, 2024. Descrição do conteúdo: pasta com os resultados dos processamento dos audio 3D. Disponível em: <https://drive.google.com/drive/folders/1iwmAPYfYiRh8SQpfcvtYR2VCvD1-sNQM?usp=sharing>. Acesso em: 5 nov. 2024.
- [30] YOST, William A.; GOUREVITCH, George. *Directional Hearing*. Springer-Verlag, New York Berlin Heidelberg London Paris Tokyo, 1987. ISBN 978-1-4612-9135-0. DOI: 10.1007/978-1-4612-4738-8.
- [31] WEIPING, Tu; RUIMIN, Hu; HENG, Wang; WENQIN, Chen. *Measurement and analysis of just noticeable difference of interaural level difference cue*. National Engineering Research Center for Multimedia Software, Wuhan University, Wuhan, China. Disponível em: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5630980>. Acesso em: 03 dez. 2024.