

Controle de Jogo com Altura Vocal: Uma Aplicação de Sinais e Sistemas

Ádrian Henrique de Abreu Paiva
Engenharia de Computação
CEFET-MG
Divinópolis, Brasil
adrianabreu@ gmail.com

Bruno Prado dos Santos
Engenharia de Computação
CEFET-MG
Divinópolis, Brasil
bruno.pradosantos1910@ gmail.com

Eduardo Henrique Queiroz Almeida	João Francisco Teles da Silva	Heitor Francisco Costa Xavier
Engenharia de Computação	Engenharia de Computação	Engenharia de Computação
CEFET-MG	CEFET-MG	CEFET-MG
Divinópolis, Brasil	Divinópolis, Brasil	Divinópolis, Brasil
eduardo.almeida@aluno.cefetmg.com	joaoteles0505@ gmail.com	heitorfrancisco2005@ hotmail.com

Resumo—Este trabalho apresenta o desenvolvimento de uma aplicação interativa que utiliza conceitos da disciplina de Sinais e Sistemas para controlar um jogo por meio da altura vocal do usuário. A proposta consiste em um jogo simples, no estilo “flappy”, onde o jogador controla a posição vertical de um personagem utilizando a intensidade de sua voz como mecanismo de entrada. A análise do sinal de áudio é feita em tempo real com técnicas de processamento digital de sinais, como a Transformada Rápida de Fourier (FFT), para extrair informações relevantes que controlam a física do jogo. A aplicação foi implementada em Python utilizando bibliotecas como `pygame`, `numpy` e `sounddevice`, sendo uma demonstração prática e lúdica dos conceitos teóricos abordados ao longo da disciplina.

Index Terms—Sinais e Sistemas, Processamento de Áudio, Controle por Voz, Transformada de Fourier, Python.

I. INTRODUÇÃO

O avanço das interfaces interativas e dos sistemas embarcados tem impulsionado o desenvolvimento de aplicações que integram processamento de sinais em tempo real com controle de dispositivos e softwares. Na área de engenharia, compreender o comportamento dos sinais e como manipulá-los é essencial para uma série de aplicações práticas, desde telecomunicações até sistemas de automação.

No contexto da disciplina de Sinais e Sistemas, este trabalho propõe o desenvolvimento de uma aplicação prática e interativa, que consiste em um jogo digital controlado pela voz do usuário. A proposta visa reforçar os conhecimentos adquiridos ao longo do curso, como a análise de sinais no domínio do tempo e da frequência, filtragem e extração de características de sinais analógicos convertidos para o domínio digital.

A ideia central do jogo é utilizar a intensidade da voz do jogador como sinal de controle. Através da análise do volume do áudio capturado em tempo real por um microfone, o sistema calcula a energia do sinal e aplica esses dados na física do jogo, controlando

o movimento vertical do personagem. Dessa forma, o jogador pode subir ou descer sua posição ao modular a força da voz, tornando a experiência mais dinâmica e divertida.

O jogo foi desenvolvido em linguagem Python, com uso das bibliotecas `pygame` para a criação da interface gráfica, `numpy` para manipulação dos dados numéricos e `sounddevice` para captura e análise do sinal de áudio. O processamento do sinal envolve cálculo de energia (RMS) e análise espectral por meio da Transformada Rápida de Fourier (FFT), ferramentas fundamentais no estudo de sinais.

II. FUNDAMENTAÇÃO TEÓRICA

A. Sinais e Sistemas

A disciplina de Sinais e Sistemas é fundamental para a engenharia, pois fornece as ferramentas conceituais para a análise e modelagem de fenômenos dinâmicos. Um **sinal** é qualquer grandeza física que varia com o tempo e que carrega informações, podendo ser representado no domínio do tempo ou da frequência. Já um **sistema** é qualquer entidade que processa sinais, modificando-os de acordo com certas regras.

Os sinais podem ser classificados em contínuos ou discretos. Neste projeto, os sinais de entrada são contínuos (áudio analógico captado pelo microfone), mas são amostrados e digitalizados para processamento computacional. Assim, trata-se de uma aplicação de sinais discretos no tempo, manipulados por um sistema digital que extrai informações e gera ações.

O estudo do comportamento de sinais em sistemas dinâmicos inclui conceitos como linearidade, causalidade, estabilidade e resposta em frequência — alguns dos quais estão presentes implicitamente neste trabalho, principalmente na forma como o sinal de áudio influencia o sistema de controle do jogo.

B. Transformada de Fourier e Análise de Áudio

Uma das ferramentas mais importantes no estudo de sinais é a **Transformada de Fourier**, que permite representar um sinal no domínio da frequência. Essa representação é especialmente útil para sinais periódicos ou complexos, como a voz humana, pois revela as componentes de frequência que compõem o sinal.

Para implementar essa análise em tempo real, é utilizada a **Transformada Rápida de Fourier (FFT)** — uma versão computacional eficiente da transformada. A FFT calcula as amplitudes de diferentes faixas de frequência a partir de blocos de sinal no tempo. Com isso, é possível visualizar o "espectro" do som captado, ou seja, quanta energia está presente em cada frequência.

No jogo desenvolvido, a FFT é usada para gerar um espectrômetro gráfico que mostra a distribuição de frequências em tempo real. Embora essa informação não influencie diretamente o controle do jogo, ela reforça visualmente a ideia de que o sinal de áudio possui conteúdo informativo que pode ser analisado e interpretado por sistemas digitais.

Além da FFT, outro conceito importante é o **valor quadrático médio** (*Root Mean Square* – RMS), que é uma medida da energia ou intensidade de um sinal. O valor de RMS é usado diretamente no jogo como indicador da "força" da voz do jogador, sendo o parâmetro de controle da movimentação do personagem.

C. Controle por Voz

O controle por voz é uma aplicação prática do processamento digital de sinais de áudio. Nessa abordagem, a voz do usuário é utilizada como um canal de entrada para um sistema interativo. Ao invés de interpretar palavras ou comandos (como em sistemas de reconhecimento de fala), este projeto se baseia apenas na **intensidade da voz**, ou seja, no volume do som captado.

Esse tipo de controle é relativamente simples de implementar, mas exige atenção à sensibilidade do sistema e ao nível de ruído do ambiente. No jogo, utiliza-se um limiar de ruído para filtrar sons muito fracos ou interferências, e a sensibilidade pode ser ajustada multiplicando o valor de RMS por um fator de ganho.

Embora seja uma abordagem básica, o controle por volume vocal demonstra claramente como sinais do mundo real (neste caso, a voz) podem ser capturados, analisados e utilizados em aplicações computacionais, conectando teoria e prática de maneira eficaz.

III. METODOLOGIA

A. Ferramentas Utilizadas

Para o desenvolvimento da aplicação foram utilizadas as seguintes ferramentas e bibliotecas:

- **Python 3:** Linguagem de programação de alto nível, utilizada por sua simplicidade e vasto ecossistema de bibliotecas.
- **pygame:** Biblioteca para desenvolvimento de jogos 2D em Python. Utilizada para a criação da interface gráfica, renderização de elementos do jogo, controle de tempo e detecção de colisões.
- **numpy:** Biblioteca para computação numérica em Python. Utilizada para manipulação de arrays, cálculos de energia do sinal (RMS) e aplicação da Transformada Rápida de Fourier (FFT).
- **sounddevice:** Biblioteca para captura de áudio em tempo real. Utilizada para acessar o microfone do sistema, capturar amostras de áudio e fornecer os dados ao sistema de controle do jogo.

As bibliotecas foram escolhidas por serem de código aberto, amplamente documentadas e de fácil integração entre si.

B. Estrutura do Jogo

O jogo desenvolvido é inspirado no estilo "flappy", no qual o jogador deve controlar a altura de um personagem para evitar obstáculos que se aproximam lateralmente. O personagem principal é representado por um quadrado que se movimenta verticalmente na tela. A cada intervalo de tempo, novos obstáculos (formados por dois blocos com um espaço entre eles) são gerados e movem-se da direita para a esquerda.

A interface gráfica foi construída com a biblioteca *pygame*, utilizando uma resolução de 800x600 pixels. O loop principal do jogo gerencia os eventos de entrada, atualiza os elementos gráficos, verifica colisões e calcula a pontuação com base na quantidade de obstáculos ultrapassados.

C. Processamento do Áudio

O controle do personagem é feito através da análise em tempo real do áudio captado pelo microfone do dispositivo. O sinal de áudio é processado da seguinte maneira:

- 1) A biblioteca *sounddevice* é usada para iniciar uma *stream* de entrada, que armazena os blocos de áudio em uma fila.
- 2) A cada ciclo do jogo, os blocos de áudio disponíveis são extraídos da fila e concatenados em um acumulador de amostras.
- 3) Quando o acumulador atinge um número suficiente de amostras (configurado como o dobro do `BLOCK_SIZE`), uma análise é realizada:
 - O valor quadrático médio (RMS) é calculado para estimar a intensidade (volume) do som captado.
 - Uma FFT é aplicada para gerar um vetor de magnitudes espectrais, que é usado para desenhar o espectrômetro na interface gráfica.
- 4) O acumulador de áudio é limpo para que o próximo bloco possa ser analisado.

A Figura Fig. 1 representa o espectrômetro gerado a partir do conteúdo de frequência da FFT, que serve como ferramenta de visualização adicional para o usuário.

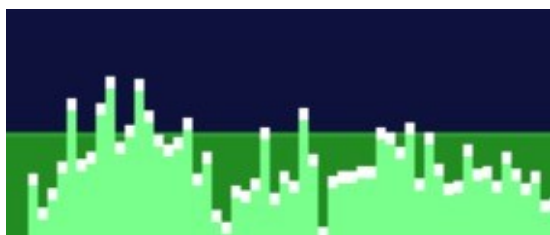


Figura 1: Espectrômetro em tempo real exibido no jogo.

D. Lógica de Controle

O movimento do personagem é regido pela física básica de gravidade e força de subida. A cada atualização do jogo:

- Uma força gravitacional é constantemente aplicada ao personagem, puxando-o para baixo.
- Quando o sinal de áudio possui um volume acima de um limiar (threshold de ruído), uma força contrária é aplicada proporcionalmente ao valor de RMS captado.
- Quanto mais forte a voz do jogador (maior RMS), maior é a força aplicada para cima.
- Esse mecanismo permite que o jogador controle a altura do personagem apenas modulando a intensidade da voz.

Além disso, o jogo conta com detecção de colisão entre o personagem e os obstáculos. Caso ocorra uma colisão, o jogo entra em estado de "fim de jogo" e aguarda uma nova entrada do teclado para reinício.

Esse modelo simples de interação demonstra, de maneira lúdica, como sinais reais podem ser usados como mecanismos de controle em tempo real em sistemas computacionais, reforçando conceitos de análise de sinais aplicados à engenharia.

IV. RESULTADOS E DISCUSSÃO

A aplicação foi testada em ambiente desktop com microfone embutido e sistema operacional Linux. O desempenho do sistema foi avaliado com foco em dois principais aspectos: a **responsividade do controle por voz** e a **estabilidade da execução em tempo real**.

Durante os testes, o jogo respondeu de forma satisfatória aos estímulos vocais do usuário. A força da voz foi corretamente interpretada como sinal de controle vertical do personagem, proporcionando uma experiência interativa divertida e desafiadora. Pequenas variações no volume resultaram em movimentos sutis, enquanto sons mais intensos geraram impulsos mais fortes, conforme o esperado.

O espectrômetro exibido na parte inferior da tela demonstrou com clareza a presença de conteúdo em diferentes frequências no sinal de voz, o que reforça

a aplicação prática da Transformada de Fourier em tempo real. Essa visualização foi útil tanto para fins didáticos quanto para auxiliar o jogador a perceber a intensidade do seu comando vocal.

A Figura Fig. 2 apresenta a tela do jogo com o espectrômetro ativo e obstáculos visíveis.



Figura 2: Tela do jogo em execução com espectrômetro ativo e obstáculos visíveis.

A Tabela I resume uma avaliação qualitativa do comportamento do sistema em diferentes situações de uso:

Tabela I: Avaliação qualitativa do sistema em diferentes condições.

Situação	Observações
Ambiente silencioso	Controle responde bem aos comandos vocais; alta sensibilidade.
Ambiente com ruído moderado	Algumas respostas indesejadas; possível necessidade de calibrar o limiar de ruído.
Ambiente barulhento	Sistema se torna instável; comandos vocais misturam-se ao ruído ambiente.
Variação do volume da voz	Movimento do personagem condizente com o esforço vocal.
Jogabilidade	Mecânica desafiadora e divertida; exige coordenação vocal e ritmo.
Estabilidade	Execução estável com taxa de quadros adequada (60 FPS).

Além dos testes subjetivos, foram realizadas simulações com diferentes volumes de fala e diferentes distâncias do microfone, com os seguintes resultados gerais:

- Distâncias curtas (10–30 cm): melhor resposta, baixa latência e boa definição do espectro.
- Distâncias médias (30–60 cm): resposta ainda boa, porém com redução na sensibilidade.
- Ambientes barulhentos (ex: ventilador ligado): movimento indesejado do personagem, exigindo modulação mais forte da voz.

Apesar dos bons resultados, algumas limitações foram observadas. Em ambientes com ruído de fundo, o sistema pode responder a sons indesejados, como falas

de outras pessoas ou ruídos mecânicos. Isso ocorre porque o controle é baseado unicamente na intensidade do sinal, sem discriminação de fonte sonora.

Além disso, usuários com dificuldade de controle vocal fino (como manter um volume constante) podem ter dificuldades em manter o personagem estável. Uma possível solução seria a adição de filtros mais robustos, como médias móveis, filtros passa-baixa ou até mesmo um sistema de reconhecimento de comandos simples.

O tempo de resposta do sistema foi considerado satisfatório para a proposta do jogo, com baixa latência entre a emissão do som e o movimento do personagem. Isso se deve à estratégia de captura em blocos pequenos (2048 amostras) e ao processamento leve utilizando a biblioteca `numpy`.

Em resumo, a aplicação cumpre seu objetivo de forma eficiente, sendo um exemplo funcional da utilização prática de conceitos de Sinais e Sistemas em uma interface interativa em tempo real.

V. CONCLUSÃO

O desenvolvimento deste projeto proporcionou uma aplicação prática e interativa dos principais conceitos abordados na disciplina de Sinais e Sistemas, especialmente no que diz respeito à análise de sinais em tempo real e à implementação de sistemas de controle baseados em entrada analógica.

Ao utilizar a voz como mecanismo de controle em um jogo digital, foi possível explorar conceitos como o valor quadrático médio (RMS), a Transformada Rápida de Fourier (FFT) e os efeitos da amostragem e discretização de sinais. A integração entre captura de áudio, processamento de sinais e resposta visual e física no ambiente gráfico demonstra, de maneira clara e lúdica, a importância do estudo de sinais em sistemas dinâmicos.

A aplicação se mostrou funcional, responsiva e interessante do ponto de vista pedagógico. Os testes indicaram um bom desempenho geral, especialmente em ambientes silenciosos. Entretanto, também foram identificadas limitações, como a sensibilidade a ruídos externos e a dificuldade de controle preciso por usuários com menor controle vocal.

Como melhorias futuras, propõe-se:

- Implementar um filtro digital (por exemplo, média móvel ou passa-baixa) para suavizar a entrada de áudio e reduzir variações bruscas.
- Adicionar um sistema de calibração automática da sensibilidade conforme o ambiente.
- Explorar novos parâmetros de controle além do volume, como frequência dominante, para ampliar a gama de interações.

Em resumo, o projeto cumpriu seu objetivo pedagógico de forma satisfatória, consolidando os conhecimentos teóricos da disciplina através de uma experiência prática, criativa e envolvente.

REFERÊNCIAS

- [1] A. V. Oppenheim, A. S. Willsky, e S. H. Nawab, *Sinais e Sistemas*, 2ª ed., Pearson, 2010.
- [2] B. P. Lathi, *Linear Systems and Signals*, 2ª ed., Oxford University Press, 2004.
- [3] W. McKinney, *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython*, 2ª ed., O'Reilly Media, 2017.
- [4] S. Van der Walt, S. C. Colbert, e G. Varoquaux, "The NumPy Array: A Structure for Efficient Numerical Computation," *Computing in Science & Engineering*, 2011.
- [5] Python Software Foundation, *Python Documentation*, disponível em: <https://docs.python.org/3/>, acesso em: 15 jul. 2025.
- [6] Pygame Organization, *Pygame Documentation*, disponível em: <https://www.pygame.org/docs/>, acesso em: 15 jul. 2025.
- [7] K. Spanderen, *SoundDevice Python Library Documentation*, disponível em: <https://python-sounddevice.readthedocs.io/>, acesso em: 15 jul. 2025.
- [8] S. W. Smith, *The Scientist and Engineer's Guide to Digital Signal Processing*, California Technical Publishing, 1997, disponível em: <http://www.dspguide.com/>, acesso em: 15 jul. 2025.