

Resumo

O trabalho apresentará um estudo comparativo entre a transformada de Hartley e a transformada de Fourier, desde seus princípios matemáticos até a amostragem prática de seus respectivos tempos de processamento, passando por detalhes de suas implementações. O ponto central deste estudo é a comparação de eficiência em termos de tempo de processamento e memória necessária para ambas as transformadas quando submetidas às condições mais semelhantes possíveis.

Introdução

Dentre os vários campos de conhecimento que são abrangidos pela engenharia da computação, o processamento digital de sinais é o que abriga o tópico abordado neste trabalho. As aplicações do processamento digital de sinais abrangem um imenso conjunto de áreas, que incluem entretenimento, comunicações, exploração do espaço, medicina, arquivologia e geofísica.

É sabido que, a transformada de Fourier, mapeia uma função do tempo em uma função da frequência. No entanto, a transformada de Fourier como definida originalmente é inviável de ser utilizada na prática, devido a seu custo computacional, sendo o número de operações da ordem de $O(n^2)$. Para resolver este problema surgiu a Transformada Rápida de Fourier (FFT) que acelerou de forma impressionante este processo, reduzindo o custo para $O(n \log 2 n)$ operações de números complexos. Por outro lado, uma solução muito semelhante surgiu como alternativa à transformada rápida de Fourier, com a vantagem de não utilizar operações aritméticas com números complexos, o que tornaria o processamento ainda mais rápido. Esta solução é denominada Transformada Rápida de Hartley (FHT). Por consequência surgiu a ideia para este trabalho, que é estudar mais a fundo os conceitos envolvidos nas duas transformadas e fazer sua comparação nas implementações práticas.

2. Trabalhos Relacionados

??

OBJETIVOS GERAIS

- Analisar desempenho de algoritmos que calculam transformada de Fourier e Hartley pela definição;
- Analisar algoritmos para computação das transformadas discretas de Fourier e Hartley, tendo como parâmetro o tempo de processamento e o custo de memória.

OBJETIVOS ESPECIFICOS

- Implementar um algoritmo de Transformada de Fourier pela definição ;
- Implementar um algoritmo de Transformada de Hartley pela definição;

- Implementar um algoritmo FFT e FHT de Cooley – Tukey base-2 e base-4;
- Implementar um algoritmo FFT e FHT split-radix;
- Analisar os dados de tempo de processamento e custo de memória para os algoritmos implementados

Justificativa

O que motiva o esforço deste trabalho é que a transformada de Fourier e Hartley têm enorme importância e um grande número de aplicações, tais como reconhecimento de padrões e processamento de sinais eletromagnéticos.

Uma vez que descobre - se um novo algoritmo que economiza custo computacional e há vários pesquisadores com implementações cujo tempo de processamento é inviável ,ocorre uma corrida para tentar usar de alguma forma este algoritmo na implementação de seus projetos. Este é um avanço que a transformadas rápidas de Fourier e Hartley possibilitam, diminuir o tempo de processamento em aplicações que variam desde resolução diferenciais ,equações integrais ,problemas inversos até incluir problemas de teoria dos números. Assim analisar o desempenho dos principais algoritmos rápidos de Fourier e Hartley e chegar em um que ,de acordo com os dados de custo e memória extraídos na pesquisa tenho o melhor desempenho, será um avanço ainda maior.

Metodologia

A execução do trabalho desta pesquisa envolverá a implementação de algoritmos computacionais que calculam a transformada de Fourier e Hartley pela definição e seus algoritmos rápidos , Cooley – Tukey base-2 e base-4 e split-radix ,totalizando assim, 8 algoritmos. . Para o desenvolvimento do projeto usar-se-á a linguagem C++, sendo o IDE utilizado, o eclipse.

As análises serão feitas em sistema operacional Windows e Linux

As ferramentas definidas para extração e monitoramento de dados no Windows serão, o gravador de desempenho do Windows e o Windows performance Analyzer, que pertencem ao Windows ADK (Kit de Avaliação e Implantação do Windows).Para SO Linux foi definida a ferramenta Google Perf Tools para o monitoramento de custo de memória e tempo.