

# Relatório Comparativo: Algoritmos de Preenchimento de Polígonos

Autor: Kaio Guilherme Ferraz de Sousa Silva

## 1. Introdução

Este relatório compara dois algoritmos de preenchimento de polígonos, explicando o funcionamento de cada um e suas principais diferenças. Para facilitar a comparação, ambos os algoritmos são utilizados para preencher o interior de um polígono definido por uma lista de vértices. O primeiro algoritmo é o **Flood Fill**, uma implementação recursiva, e o segundo é o **Varredura Geométrica** (Scan Line), que utiliza uma abordagem de varredura linha por linha.

### Algoritmos Analisados

1. **Flood Fill**: Algoritmo recursivo simples para preenchimento de regiões contidas em um polígono.
  2. **Varredura Geométrica**: Algoritmo que realiza o preenchimento do polígono linha por linha, utilizando uma tabela de lados.
- 

## 2. Descrição do Funcionamento

### 2.1 Algoritmo de Flood Fill

O algoritmo **Flood Fill** começa em um ponto interno do polígono e realiza uma busca recursiva nos quatro vizinhos (direita, esquerda, acima e abaixo). Ele verifica se o pixel atual possui a cor alvo (a cor a ser preenchida) e substitui por uma nova cor. A recursão continua enquanto houver pixels com a cor alvo a serem preenchidos.

- **Processo:**

1. Começa com um ponto interno dentro do polígono.

2. Verifica os pixels adjacentes (superior, inferior, à esquerda, à direita).
3. Se o pixel for da cor alvo, ele é alterado para a nova cor.
4. A recursão é realizada até que todos os pixels dentro da área sejam preenchidos.

## 2.2 Algoritmo de Varredura Geométrica

O algoritmo de **Varredura Geométrica** realiza o preenchimento do polígono linha por linha, mas ao contrário do **Flood Fill**, ele começa da linha de varredura mais alta até a mais baixa. Ele calcula as interseções das arestas do polígono com cada linha de varredura e, utilizando a regra de paridade, decide quais regiões entre as interseções devem ser preenchidas.

- **Processo:**

1. Constrói uma tabela de lados para armazenar as arestas do polígono.
2. Varre a tela da linha de varredura mais alta para a mais baixa, encontrando as interseções da linha de varredura com as arestas do polígono.
3. Ordena as interseções e aplica a regra de paridade para determinar quais segmentos devem ser preenchidos.
4. Preenche os pixels entre as interseções.

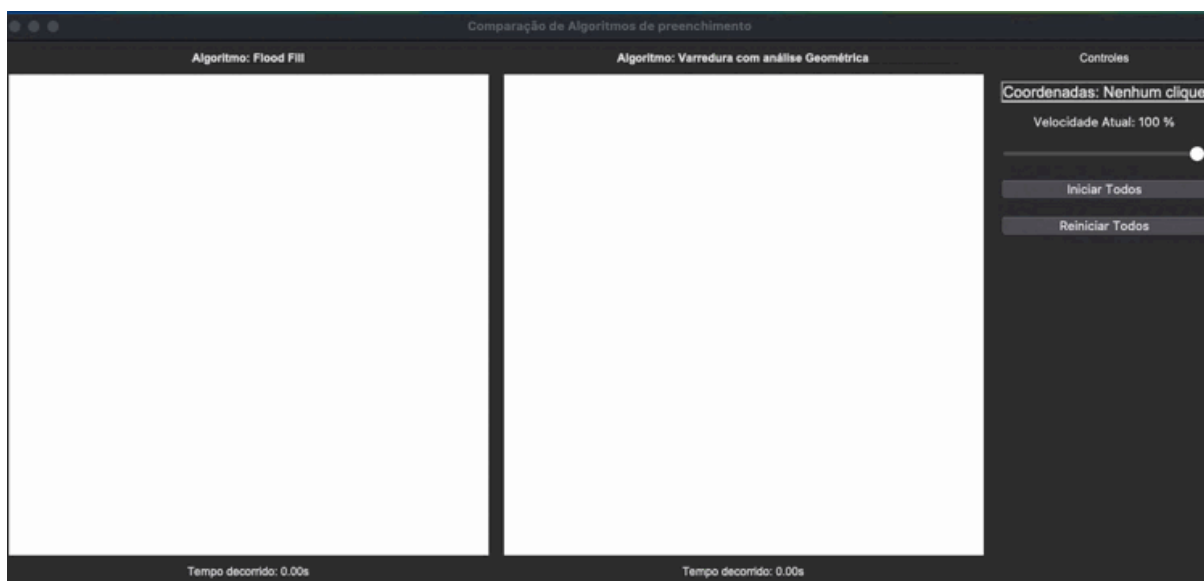
## 3. Diferenças Entre os Algoritmos

Característica	<b>Flood Fill</b>	<b>Varredura Geométrica</b>
<b>Tipo de Algoritmo</b>	Recursivo, baseado em busca em profundidade.	Iterativo, baseado em varredura linha por linha.
<b>Uso de Memória</b>	Utiliza a pilha de recursão, o que pode causar problemas com áreas grandes.	Utiliza memória para armazenar a tabela de lados.
<b>Velocidade</b>	Pode ser mais lento, especialmente em áreas grandes devido à recursão.	Geralmente mais rápido, pois preenche linhas inteiras de uma vez.
<b>Preenchimento de Áreas</b>	Preenche regiões conectadas a partir do ponto inicial.	Preenche áreas com base na análise geométrica das interseções.

<b>Limitações</b>	Pode ter problemas em áreas grandes devido à profundidade da recursão.	Funciona bem em áreas complexas com contornos bem definidos.
-------------------	--	--

## 4. Análise de Desempenho

Os algoritmos **Flood Fill** e **Varredura Geométrica** são executados simultaneamente, cada um em uma thread separada, garantindo que o desempenho de um não interfira no outro. O algoritmo Flood Fill realiza o preenchimento de um polígono simples a partir de um ponto interno, utilizando recursão para percorrer e colorir a área de maneira contínua. Já o algoritmo de Varredura Geométrica preenche o polígono de maneira sistemática, calculando as interseções entre as arestas e preenchendo os pixels entre elas de cima para baixo. A execução independente dos algoritmos pode ser visualizada no GIF abaixo, permitindo comparar suas particularidades e desempenhos.



Foram usadas matrizes de 500×500 pixel em escala 1:1

## 5. Conclusão

Ambos os algoritmos têm suas vantagens e desvantagens. O **Flood Fill** é simples de implementar e ideal para regiões pequenas ou com pouca complexidade, mas sua recursão pode ser um problema em polígonos grandes. O **Varredura Geométrica**, por outro lado, é mais eficiente para polígonos complexos e grandes, realizando o preenchimento de maneira mais controlada

e com um uso de memória mais otimizado. A escolha do algoritmo depende do tipo de aplicação e das características do polígono a ser preenchido.